NON-INTRUSIVE REDUCED-ORDER MODELS FOR PARAMETRIC PARTIAL DIFFERENTIAL EQUATIONS VIA DATA-DRIVEN OPERATOR INFERENCE

SHANE A. MCQUARRIE*, PARISA KHODABAKHSHI*, AND KAREN E. WILLCOX*

Abstract. This work formulates a new approach to reduced modeling of parameterized, time-dependent partial differential equations (PDEs). The method employs Operator Inference, a scientific machine learning framework combining data-driven learning and physics-based modeling. The parametric structure of the governing equations is embedded directly into the reduced-order model, and parameterized reduced-order operators are learned via a data-driven linear regression problem. The result is a reduced-order model that can be solved rapidly to map parameter values to approximate PDE solutions. Such parameterized reduced-order models may be used as physics-based surrogates for uncertainty quantification and inverse problems that require many forward solves of parametric PDEs. Numerical issues such as well-posedness and the need for appropriate regularization in the learning problem are considered, and an algorithm for hyperparameter selection is presented. The method is illustrated for a parametric heat equation and demonstrated for the FitzHugh-Nagumo neuron model.

Key words. Parametric model reduction, operator inference, scientific machine learning, data-driven reduced model

AMS subject classifications. 35B30, 35R30, 65F22

1. Introduction.

Model reduction seeks to alleviate the computational burden of large-scale numerical simulations of dynamical systems by constructing reduced-order models (ROMs) that accurately capture the system dynamics, but which are much less expensive to solve than the high-fidelity models inherent in applications. The challenge is to generate ROMs from limited training data that respond well to changes in the scenario parameters that define the governing dynamics [9]. Such parametric ROMs are critical for enabling outer-loop applications such as design, inverse problems, optimization, and uncertainty quantification. Furthermore, as high-fidelity simulations become increasingly sophisticated and simulation data becomes more available, there is a growing need for non-intrusive model reduction methods, which aim to learn ROMs primarily from simulation data and/or outputs, as opposed to making a direct reduction of the underlying high-fidelity code that produced them [17]. Non-intrusive approaches combine data-driven learning with physics-based modeling in a way that enables both flexibility and robustness. This paper presents a framework for learning parametric ROMs in a non-intrusive fashion.

Adapting non-intrusive model reduction strategies to the parametric setting is an active area of research. One major model reduction strategy, Dynamic mode decomposition (DMD) [14, 40], learns a low-dimensional linear mapping based on state space data, approximating the eigenstates of the infinite-dimensional Koopman operator. The work in [43] targets parametric problems by incorporating DMD with an active subspace strategy to reduce the dimensionality of the parameter space. Methods based on the Loewner framework [2, 3], another common non-intrusive approach to model reduction, build a ROM based on input-output measurements and transfer functions. Loewner methods have been generalized to parameterized linear systems by introducing additional degrees of freedom in the construction of the reduced-order transfer function to account for parametric dependencies [21]. Recent work in [11]

*Oden Institute for Computational Engineering and Sciences, University of Texas at Austin, USA (shanemcq@utexas.edu, parisa@austin.utexas.edu, kwillcox@oden.utexas.edu).

blends ideas from classical model reduction with deep learning to construct a mapping from parametric inputs to state outputs. Deep learning approaches to model reduction aim to benefit from the flexibility of representing the state on a low-dimensional but nonlinear manifold [26].

Equation discovery methods, in which the governing equations of a dynamical system are learned from data, share some characteristics with non-intrusive model reduction methods. One class of equation discovery approaches uses sparse regression to identify the underlying PDEs from a set of data [39] or the key terms in a dynamical system within a library of potential nonlinear terms [13, 36, 38]. In a similar vein, the work in [4, 27] combines machine learning techniques with manifold learning algorithms to learn a macroscopic model for long-wavelength behavior corresponding to fine-scale measurement data. Each of these methods relies on an appropriate candidate library for the terms of the unknown equations, selecting the best combination based on high-dimensional data. In contrast, model reduction approaches seek a low-dimensional representation of a system, hence the learning is typically done in a reduced space.

Operator Inference (OpInf), introduced in [34], is a non-intrusive framework for model reduction of systems with polynomial nonlinearities. As with other non-intrusive approaches, the method does not require intrusive access to source code, instead inferring the ROM solely from initial conditions, simulation snapshots, and corresponding inputs. Known governing equations motivate the form of the ROM, and the operators defining the ROM are chosen by minimizing a data-driven residual in a reduced state space. The associated learning problem is linear, dense, relatively small, and has a closed-form solution.

Since its introduction, the OpInf framework has been expanded in several ways: transforming variables (lifting) to induce the requisite polynomial structure [35, 42]; approximating nonpolynomial nonlinearities via the discrete empirical interpolation method (DEIM) [8]; regularizing the learning problem to enable performance on large-scale systems [22, 29]; re-projecting trajectories to exactly recover intrusive ROMs [33]; and accounting for algebraic equations arising from lifting transformations [24], to name a few. In these OpInf-based methods, parametric dependencies are addressed by learning separate ROMs for individual parameter samples, then interpolating either their reduced operators [33, 34] or their outputs [24]. In this paper, we show that the parametric structure of the governing equations of interest can be built directly into the OpInf regression problem, circumventing the need for interpolation, if the parametric dependencies have an affine form. Our contribution enables ROMs for affine parametric systems with multiple parameters since interpolation in more than one or two parameter dimensions quickly becomes challenging. Affine parametric problems have been studied frequently in the context of reduced basis methods [20, 37, 48, 49], in which the preservation of the affine parametric structure by projection plays a key role [9]. The parametric structure preservation was also exploited in a recent OpInf work applied to the shallow water equations in non-traditional form [50]. The approach presented here includes the following key contributions: 1) formulating a general framework for parametric OpInf in a time-continuous setting, 2) implementing a robust regularization strategy in the learning problem, and 3) establishing a priori conditions for determining the well-posedness of the inference procedure.

The remainder of the paper is organized as follows. Section 2 establishes the general methodology; Section 3 extends the framework to systems of PDEs; Section 4 details the computational aspects of solving the parametric OpInf problem; Section 5 presents two numerical examples; and Section 6 concludes the paper.

2. Non-intrusive Parametric Model Reduction. In subsection 2.1, we show how the form of an appropriate ROM can be determined directly from the form of certain PDEs; subsection 2.2 presents the OpInf approach for learning such ROMs from data and system structure. We also introduce a heat equation example for which numerical results are reported in subsection 5.1.

2.1. Projection-based Reduced-order Models of Parametric PDEs. We target systems governed by parametric PDEs that are polynomial in state, which includes linear PDEs as well as a large class of nonlinear PDEs. Let $\Omega \subset \mathbb{R}^{d_x}$ be an open, bounded set with Lipschitz continuous boundary $\partial\Omega = \Gamma \cup (\partial\Omega \backslash \Gamma)$ and outward-pointing normal $\boldsymbol{\eta} \in \mathbb{R}^{d_x}$. For the time domain $[t_0, t_f] \subset \mathbb{R}$ and the parameter domain $\mathcal{P} \subset \mathbb{R}^{d_\mu}$, we consider the initial/boundary-value problem

$$\frac{\partial u}{\partial t} = \mathcal{F}(u;\mu), \qquad x \in \Omega,\ t \in (t_0, t_f],\ \mu \in \mathcal{P}, \tag{2.1a}$$

$$u(x, t_0; \mu) = u_0(x;\mu), \qquad x \in \Omega,\ \mu \in \mathcal{P}, \tag{2.1b}$$

$$u(x, t; \mu) = 0, \qquad x \in \Gamma,\ t \in [t_0, t_f],\ \mu \in \mathcal{P}, \tag{2.1c}$$

$$\boldsymbol{\eta} \cdot \nabla_x u(x, t; \mu) = 0, \qquad x \in \partial\Omega \setminus \Gamma,\ t \in [t_0, t_f],\ \mu \in \mathcal{P}, \tag{2.1d}$$

where the unknown state variable $u(\cdot, t; \mu)$ is contained in a separable Hilbert space $\mathcal{V}$ of real-valued functions satisfying the boundary conditions (2.1c)–(2.1d) with dual space $\mathcal{V}^*$, and $\mathcal{F} : \mathcal{V} \times \mathcal{P} \to \mathcal{V}^*$ is a spatial differential operator that depends on the free parameter $\mu \in \mathcal{P}$. We assume (2.1a)–(2.1d) has a unique solution u in at least the weak sense, meaning

$$\left\langle v, \frac{\partial u}{\partial t} \right\rangle = \langle v, \mathcal{F}(u;\mu) \rangle \qquad \text{for all} \qquad v \in \mathcal{V},\ t \in [t_0, t_f],\ \mu \in \mathcal{P}, \tag{2.2}$$

where $\langle \cdot, \cdot \rangle$ is the duality pairing of $\mathcal{V}$ with $\mathcal{V}^*$.

We consider the setting in which $\mathcal{F}$ has a polynomial structure with respect to the state u and its spatial derivatives. Many PDEs enjoy this structure or can be written in this form through a change of variables [35]. For brevity we consider a quadratic form, but higher-order (e.g., cubic) terms may also be included, as we will see later in Example 3.1. Specifically, suppose

$$\mathcal{F}(u;\mu) = \mathcal{C}(\mu) + \mathcal{A}(u;\mu) + \mathcal{H}(u, u;\mu), \tag{2.3a}$$

where $\mathcal{C} : \mathcal{P} \to \mathcal{V}^*$, and where $\mathcal{A} : \mathcal{V} \times \mathcal{P} \to \mathcal{V}^*$ and $\mathcal{H} : \mathcal{V} \times \mathcal{V} \times \mathcal{P} \to \mathcal{V}^*$ are linear in each of their state arguments. Furthermore, assume that the operators $\mathcal{C}$, $\mathcal{A}$, and $\mathcal{H}$ exhibit the following affine decompositions with respect to the parameter μ:

$$\begin{aligned} \mathcal{C}(\mu) &= \sum_{p=1}^{q_c} \theta_c^{(p)}(\mu)\mathcal{C}^{(p)}, \qquad \mathcal{A}(u;\mu) = \sum_{p=1}^{q_A} \theta_A^{(p)}(\mu)\mathcal{A}^{(p)}(u), \\ \mathcal{H}(u, v;\mu) &= \sum_{p=1}^{q_H} \theta_H^{(p)}(\mu)\mathcal{H}^{(p)}(u, v), \end{aligned} \tag{2.3b}$$

where $\mathcal{C}^{(p)} \in \mathcal{V}^*$, $\mathcal{A}^{(p)} : \mathcal{V} \to \mathcal{V}^*$, $\mathcal{H}^{(p)} : \mathcal{V} \times \mathcal{V} \to \mathcal{V}^*$, and the scalar-valued functions $\theta_c^{(p)}, \theta_A^{(p)}, \theta_H^{(p)} : \mathcal{P} \to \mathbb{R}$ are such that the sets $\{\theta_c^{(p)}\}_{p=1}^{q_c}$, $\{\theta_A^{(p)}\}_{p=1}^{q_A}$, and $\{\theta_H^{(p)}\}_{p=1}^{q_H}$ are each linearly independent. Note that the operators $\mathcal{C}^{(p)}$, $\mathcal{A}^{(p)}$,and $\mathcal{H}^{(p)}$ are independent of the parameter μ. The affine structure either occurs naturally in the PDE or can be created via approximation with the empirical interpolation method [7].

A projection-based ROM of (2.1a)–(2.1d) with $\mathcal{F}$ as in (2.3a)–(2.3b) retains the affine-parametric polynomial structure of the system [9, 19]. Let $\{v_j\}_{j=1}^{\infty} \subset \mathcal{V}$ be an orthonormal set such that the solution u may be expressed with the expansion

$$u(x,t;\mu) = \sum_{j=1}^{\infty} \hat{u}_j(t;\mu) v_j(x). \tag{2.4}$$

Since $\langle v_i, v_j \rangle = \delta_{ij}$, the coefficients satisfy $\hat{u}_j(t;\mu) = \langle v_j, u(\cdot,t;\mu) \rangle$. A reduced model with $r \in \mathbb{N}$ degrees of freedom consists of time evolution equations for the coefficients $\hat{u}_1(t;\mu), \ldots, \hat{u}_r(t;\mu)$; the approximate ROM solution $\breve{u}$ of (2.1a)–(2.1d) is then given by the sum (2.4), truncated to r terms:

$$\breve{u}(x,t;\mu) = \sum_{j=1}^{r} \hat{u}_j(t;\mu) v_j(x). \tag{2.5}$$

Note that $\breve{u}$ is confined to the finite-dimensional subspace $\operatorname{span}(\{v_1, \ldots, v_r\}) \subset \mathcal{V}$. By substituting $\breve{u}$ for u in (2.2) with test function $v = v_i$, and using the form of $\mathcal{F}$ from (2.3a)–(2.3b), we obtain

$$\begin{aligned} \frac{\mathrm{d}\hat{u}_i}{\mathrm{d}t} = &\sum_{p=1}^{q_c} \theta_c^{(p)}(\mu) \left\langle v_i, \mathcal{C}^{(p)}(\mu) \right\rangle + \sum_{p=1}^{q_A} \theta_A^{(p)}(\mu) \sum_{j=1}^{r} \left\langle v_i, \mathcal{A}^{(p)}\left(v_j;\mu\right) \right\rangle \hat{u}_j \\ &+ \sum_{p=1}^{q_H} \theta_H^{(p)}(\mu) \sum_{j=1}^{r} \sum_{k=1}^{r} \left\langle v_i, \mathcal{H}^{(p)}\left(v_j, v_k;\mu\right) \right\rangle \hat{u}_j \hat{u}_k. \end{aligned} \tag{2.6}$$

Collecting (2.6) for $i = 1, \ldots, r$ yields a uniquely-defined system of ordinary differential equations (ODEs) with state vector $\widehat{\mathbf{u}}(t;\mu) = \begin{bmatrix} \hat{u}_1(t;\mu) & \cdots & \hat{u}_r(t;\mu) \end{bmatrix}^{\top} \in \mathbb{R}^r$,

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{u}}(t;\mu) = &\left(\sum_{p=1}^{q_c} \theta_c^{(p)}(\mu) \widehat{\mathbf{c}}^{(p)} \right) + \left(\sum_{p=1}^{q_A} \theta_A^{(p)}(\mu) \widehat{\mathbf{A}}^{(p)} \right) \widehat{\mathbf{u}}(t;\mu) \\ &+ \left(\sum_{p=1}^{q_H} \theta_H^{(p)}(\mu) \widehat{\mathbf{H}}^{(p)} \right) \left(\widehat{\mathbf{u}}(t;\mu) \mathbin{\widehat{\odot}} \widehat{\mathbf{u}}(t;\mu) \right), \end{aligned} \tag{2.7a}$$

$$\widehat{\mathbf{u}}(t_0;\mu) = \begin{bmatrix} \langle v_1, u_0(\mu) \rangle & \cdots & \langle v_r, u_0(\mu) \rangle \end{bmatrix}^{\top} \in \mathbb{R}^r, \tag{2.7b}$$

where $\widehat{\mathbf{c}}^{(p)} \in \mathbb{R}^r$, $\widehat{\mathbf{A}}^{(p)} \in \mathbb{R}^{r \times r}$, $\widehat{\mathbf{H}}^{(p)} \in \mathbb{R}^{r \times \binom{r+1}{2}}$, and $\mathbin{\widehat{\odot}}$ denotes a compact Khatri-Rao product, i.e., $\mathbf{u} \mathbin{\widehat{\odot}} \mathbf{u}$ extracts the unique components of the Khatri-Rao product of $\mathbf{u}$ with itself (see Appendix A). Note that (2.7a) and (2.3a)–(2.3b) are both quadratic in their respective state and affine-parametric with respect to μ, with the same functions $\theta_c^{(p)}$, $\theta_A^{(p)}$, and $\theta_H^{(p)}$ in the affine expansions.

The system (2.7a)–(2.7b) is a ROM for (2.1a)–(2.1d) in which the boundary conditions are embedded through the basis functions. The quality of the ROM as a surrogate for (2.1a)–(2.1d) depends heavily on the basis $\{v_j\}_{j=1}^{r}$, but the *form* of the equations is the same for any choice of basis. In subsection 2.2, we leverage this property to develop a procedure for learning such a model without explicitly evaluating the terms in (2.6).

REMARK 2.1. *As written above, (2.7a) is the most general form of a quadratic ODE with affine-parametric structure as in (2.3b). However, as we will see in examples, the constant, linear, and quadratic terms are not always present in practice, and when they are the number of terms in the affine expansions (i.e. q_c, q_A, and q_H) tends to be small. In other words, projection-based ROMs of the form (2.7a) have the same number of nonzero terms as the original PDE (2.1a).*

EXAMPLE 2.2 (Heat Equation). *Consider the one-dimensional spatial domain $\Omega = (0,1) \subset \mathbb{R}$ and the parameter domain $\mathcal{P} = [.01, 2.5] \times [0.1, 2.5] \subset \mathbb{R}^2$. For a fixed $\bar{x} \in \Omega$, let $\chi_{[0,\bar{x})}$ and $\chi_{[\bar{x},1]}$ be indicator functions over Ω,*

$$\chi_{[0,\bar{x})}(x) = \begin{cases} 1, & 0 < x < \bar{x}, \\ 0, & \bar{x} \leq x < 1, \end{cases} \qquad \chi_{[\bar{x},1]}(x) = \begin{cases} 0, & 0 < x < \bar{x}, \\ 1, & \bar{x} \leq x < 1. \end{cases}$$

The following equation models the diffusion of heat through a one-dimensional rod composed of two materials with distinct thermal diffusivities $\mu = (\alpha, \beta) \in \mathcal{P}$, with the temperature prescribed at each end of the rod:

$$\frac{\partial u}{\partial t} = \left(\alpha \chi_{[0,\bar{x})} + \beta \chi_{[\bar{x},1]}\right) \frac{\partial^2 u}{\partial x^2} \qquad x \in \Omega,\ t \in (t_0, t_f],\ \mu \in \mathcal{P}, \tag{2.8a}$$

$$u(x, t_0; \mu) = u_0(x; \mu), \qquad x \in \Omega,\ \mu \in \mathcal{P}, \tag{2.8b}$$

$$u(0, t; \mu) = u(1, t; \mu) = 0, \qquad t \in [t_0, t_f],\ \mu \in \mathcal{P}. \tag{2.8c}$$

The underlying Hilbert space containing the state u is $\mathcal{V} = H^2(\Omega) \cap H_0^1(\Omega)$, the set of twice weakly differentiable functions satisfying the homogeneous Dirichlet boundary conditions (2.8c). In the language of (2.3a)–(2.3b), $\mathcal{C} \equiv \mathcal{H} \equiv 0$ and

$$\mathcal{A}(u; \mu) = \theta_A^{(1)}(\mu)\mathcal{A}^{(1)}(u) + \theta_A^{(2)}(\mu)\mathcal{A}^{(2)}(u)$$

with $\theta_A^{(1)}(\mu) = \alpha$, $\theta_A^{(1)}(\mu) = \beta$, $\mathcal{A}^{(1)}(u) = \chi_{[0,\bar{x})} \frac{\partial^2 u}{\partial x^2}$, and $\mathcal{A}^{(2)}(u) = \chi_{[\bar{x},1]} \frac{\partial^2 u}{\partial x^2}$. A projection-based ROM for this problem as in (2.7a) has the form

$$\frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{u}}(t; \mu) = \left(\alpha \mathbf{A}^{(1)} + \beta \mathbf{A}^{(2)}\right) \widehat{\mathbf{u}}(t; \mu), \tag{2.9}$$

with the initial condition as in (2.7b).

2.2. Affine Operator Inference for PDEs. Constructing $\widehat{\mathbf{c}}^{(p)}$, $\widehat{\mathbf{A}}^{(p)}$, and $\widehat{\mathbf{H}}^{(p)}$ in the ODE system (2.7a) by evaluating the terms of (2.6) is an inherently intrusive process, requiring explicit access to the differential operators $\mathcal{C}^{(p)}$, $\mathcal{A}^{(p)}$, and $\mathcal{H}^{(p)}$. In this section, we construct affine-parametric Operator Inference to *learn* the ROM through a data-driven optimization problem, given a basis and observations of the solution u.

Consider (2.7a) with fixed, known integers $q_c, q_A, q_H \geq 0$ and affine coefficient functions $\theta = \{\theta_c^{(1)}, \ldots, \theta_H^{(q_H)}\}$, which define the polynomial and affine-parametric structure of the system. Define

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{u}}(t; \mu) = \mathbf{F}(\widehat{\mathbf{O}}; \widehat{\mathbf{u}}, t, \theta, \mu) = &\left(\sum_{p=1}^{q_c} \theta_c^{(p)}(\mu)\widehat{\mathbf{c}}^{(p)}\right) + \left(\sum_{p=1}^{q_A} \theta_A^{(p)}(\mu)\widehat{\mathbf{A}}^{(p)}\right) \widehat{\mathbf{u}}(t; \mu) \\ &+ \left(\sum_{p=1}^{q_H} \theta_H^{(p)}(\mu)\widehat{\mathbf{H}}^{(p)}\right) \left(\widehat{\mathbf{u}}(t; \mu) \mathbin{\widehat{\odot}} \widehat{\mathbf{u}}(t; \mu)\right), \end{aligned} \tag{2.10a}$$

where the as yet unknown *operator matrix* $\widehat{\mathbf{O}}$ is the concatenation

$$\widehat{\mathbf{O}} = \left[\begin{array}{c|c|c} \widehat{\mathbf{c}}^{(1)} \ \cdots \ \widehat{\mathbf{c}}^{(q_c)} & \widehat{\mathbf{A}}^{(1)} \ \cdots \ \widehat{\mathbf{A}}^{(q_A)} & \widehat{\mathbf{H}}^{(1)} \ \cdots \ \widehat{\mathbf{H}}^{(q_H)} \end{array}\right] \in \mathbb{R}^{r \times q(r)}, \tag{2.10b}$$

with column dimension $q(r) = q_c + q_A r + q_H \binom{r+1}{2}$. Equipped with the initial condition (2.7b), (2.10a)–(2.10b) describe the family of ROMs with the same polynomial and parametric form as (2.7a), with particular realizations determined by the operator matrix $\widehat{\mathbf{O}}$. The goal is to choose $\widehat{\mathbf{O}}$ such that the corresponding ROM accurately captures the dynamics of the governing PDE (2.1a)–(2.1d) for all $\mu \in \mathcal{P}$.

We learn $\widehat{\mathbf{O}}$ by solving a data-driven least-squares regression problem. Suppose we can sample the solution of (2.1a)–(2.1d) at s parameter values $\{\mu_i\}_{i=1}^s \subset \mathcal{P}$ and K times $\{t_j\}_{j=1}^K \subset [t_0, t_f]$. We define the loss function $\mathcal{L} : \mathbb{R}^{r \times q(r)} \to \mathbb{R}$ associated with these solution samples to be the sum of the residuals of (2.10a),

$$\mathcal{L}(\widehat{\mathbf{O}}) = \sum_{i=1}^{s} \sum_{j=1}^{K} \left\| \mathbf{F}(\widehat{\mathbf{O}}; \widehat{\mathbf{u}}, t_j, \theta, \mu_i) - \frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{u}}(t;\mu_i)\Big|_{t=t_j} \right\|_2^2, \tag{2.11}$$

where $\widehat{\mathbf{u}}(t;\mu) = [\langle v_1, u(\cdot,t;\mu)\rangle \ \cdots \ \langle v_r, u(\cdot,t;\mu)\rangle]^\top \in \mathbb{R}^r$ as before, and with $\mathbf{F}$ and $\widehat{\mathbf{O}}$ as in (2.10a)–(2.10b). To write the minimization of (2.11) in a standard form, define the matrices

$$\begin{aligned} \widehat{\mathbf{U}}(\mu_i) &= \left[\begin{array}{ccc} \widehat{\mathbf{u}}(t_1;\mu_i) & \cdots & \widehat{\mathbf{u}}(t_K;\mu_i) \end{array}\right] \in \mathbb{R}^{r \times K}, \\ \dot{\widehat{\mathbf{U}}}(\mu_i) &= \left[\begin{array}{ccc} \frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{u}}(t;\mu_i)\Big|_{t=t_1} & \cdots & \frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{u}}(t;\mu_i)\Big|_{t=t_K} \end{array}\right] \in \mathbb{R}^{r \times K}, \end{aligned}$$

which group the projected solution and the associated time derivatives for each parameter sample, and define the row vectors

$$\begin{aligned} \boldsymbol{\theta}_c(\mu_i) &= \left[\begin{array}{ccc} \theta_c^{(1)}(\mu_i) & \cdots & \theta_c^{(q_c)}(\mu_i) \end{array}\right] \in \mathbb{R}^{1 \times q_c}, \\ \boldsymbol{\theta}_A(\mu_i) &= \left[\begin{array}{ccc} \theta_A^{(1)}(\mu_i) & \cdots & \theta_A^{(q_A)}(\mu_i) \end{array}\right] \in \mathbb{R}^{1 \times q_A}, \\ \boldsymbol{\theta}_H(\mu_i) &= \left[\begin{array}{ccc} \theta_H^{(1)}(\mu_i) & \cdots & \theta_H^{(q_H)}(\mu_i) \end{array}\right] \in \mathbb{R}^{1 \times q_H}, \end{aligned}$$

which encode the affine parametric dependencies of $\mathcal{C}$, $\mathcal{A}$, and $\mathcal{H}$, respectively. Finally, define the *data matrix*

$$\begin{aligned} \mathbf{D} &= \left[\begin{array}{c|c|c} \mathbf{D}_c & \mathbf{D}_A & \mathbf{D}_H \end{array}\right] \\ &= \left[\begin{array}{c|c|c} \boldsymbol{\theta}_c(\mu_1) \otimes \mathbf{1}_K & \boldsymbol{\theta}_A(\mu_1) \otimes \widehat{\mathbf{U}}(\mu_1)^\top & \boldsymbol{\theta}_H(\mu_1) \otimes \left(\widehat{\mathbf{U}}(\mu_1) \mathbin{\widehat{\odot}} \widehat{\mathbf{U}}(\mu_1)\right)^\top \\ \vdots & \vdots & \vdots \\ \boldsymbol{\theta}_c(\mu_s) \otimes \mathbf{1}_K & \boldsymbol{\theta}_A(\mu_s) \otimes \widehat{\mathbf{U}}(\mu_s)^\top & \boldsymbol{\theta}_H(\mu_s) \otimes \left(\widehat{\mathbf{U}}(\mu_s) \mathbin{\widehat{\odot}} \widehat{\mathbf{U}}(\mu_s)\right)^\top \end{array}\right], \end{aligned} \tag{2.12}$$

where $\mathbf{1}_K \in \mathbb{R}^K$ is a column vector of unity of length K. We then have

$$\text{pOpInf:} \qquad \min_{\widehat{\mathbf{O}}} \mathcal{L}(\widehat{\mathbf{O}}) = \min_{\widehat{\mathbf{O}}} \left\| \mathbf{D}\widehat{\mathbf{O}}^\top - \mathbf{R}^\top \right\|_F^2, \tag{2.13}$$

where $\mathbf{D} \in \mathbb{R}^{sK \times q(r)}$, $\widehat{\mathbf{O}} \in \mathbb{R}^{r \times q(r)}$, and $\mathbf{R} = [\dot{\widehat{\mathbf{U}}}(\mu_1) \ \cdots \ \dot{\widehat{\mathbf{U}}}(\mu_s)] \in \mathbb{R}^{r \times sK}$. We call (2.13) the *affine-parametric Operator Inference problem* (pOpInf). Note that (2.13)

is a linear least-squares problem which decouples by the columns of $\widehat{\mathbf{O}}^\top$, meaning the dynamics for each $\hat{u}_i$ are learned independently [34].

If the data matrix $\mathbf{D}$ has full column rank, then (2.13) has a unique closed-form solution [12, 18, 45]. However, $\mathbf{D}$ is susceptible to rank deficiencies due to its Kronecker block structure. Theorem 2.3 establishes necessary conditions for the well-posedness of the pOpInf problem (2.13) and sufficient conditions in the monomial case (e.g., $\mathcal{C} = \mathcal{A} = 0$ but $\mathcal{H} \neq 0$ so that $\mathbf{D} = \mathbf{D}_H$).

THEOREM 2.3. *Let $\Theta_c \in \mathbb{R}^{s\times q_c}$, $\Theta_A \in \mathbb{R}^{s\times q_A}$, and $\Theta_H \in \mathbb{R}^{s\times q_H}$ be the matrices with entries*

$$[\Theta_c]_{ij} = \theta_c^{(j)}(\mu_i), \qquad [\Theta_A]_{ij} = \theta_A^{(j)}(\mu_i), \qquad [\Theta_H]_{ij} = \theta_H^{(j)}(\mu_i), \tag{2.14}$$

that is, the ith row of Θ_c is $\boldsymbol{\theta}_c(\mu_i)$, the ith row of Θ_A is $\boldsymbol{\theta}_A(\mu_i)$, and the ith row of Θ_H is $\boldsymbol{\theta}_H(\mu_i)$. If any of Θ_c, Θ_A, or Θ_H do not have full column rank, then the data matrix $\mathbf{D} = [\mathbf{D}_c \ \mathbf{D}_A \ \mathbf{D}_H]$ defined in (2.12) is rank deficient. Furthermore, if either of the block matrices

$$\widehat{\mathbf{U}}_A = \begin{bmatrix} \widehat{\mathbf{U}}(\mu_1)^\top \\ \vdots \\ \widehat{\mathbf{U}}(\mu_s)^\top \end{bmatrix}, \qquad \widehat{\mathbf{U}}_H = \begin{bmatrix} \left(\widehat{\mathbf{U}}(\mu_1) \mathbin{\widehat{\odot}} \widehat{\mathbf{U}}(\mu_1)\right)^\top \\ \vdots \\ \left(\widehat{\mathbf{U}}(\mu_s) \mathbin{\widehat{\odot}} \widehat{\mathbf{U}}(\mu_s)\right)^\top \end{bmatrix}$$

do not have full column rank, then neither does $\mathbf{D}$. On the other hand, if Θ_A and each $\widehat{\mathbf{U}}(\mu_i)^\top$ $(i = 1, \ldots, s)$ have full column rank, then so does $\mathbf{D}_A$; and if Θ_H and each $(\widehat{\mathbf{U}}(\mu_i) \mathbin{\widehat{\odot}} \widehat{\mathbf{U}}(\mu_i))^\top$ have full column rank, then so does $\mathbf{D}_H$.

Proof. Note that $\Theta_c = \mathbf{D}_c$, the leftmost block of $\mathbf{D}$. Therefore, if Θ_c does not have full column rank, then $\mathbf{D}$ has linearly dependent columns. Lemma B.1 gives the result for the remaining cases: if Θ_A or $\widehat{\mathbf{U}}_A$ do not have full column rank, apply Lemma B.1 with $\mathbf{y}_i = \boldsymbol{\theta}_A(\mu_i)^\top$, $\mathbf{Z}_i = \widehat{\mathbf{U}}(\mu_i)^\top$, and $\mathbf{W} = \mathbf{D}_A$ to show $\mathbf{D}_A$ is rank deficient; a similar argument holds for Θ_H and $\widehat{\mathbf{U}}_H$, showing $\mathbf{D}_H$ is rank deficient. The results to the converse also hold by Lemma B.1. ∎

Selecting appropriate parameter samples is an important issue for all parametric model reduction methods [9]. Theorem 2.3 provides numerically relevant guidance for parameter selection in the context of (2.13): for a set of parameter values $\{\mu_i\}_{i=1}^s$, the (small) matrices Θ_c, Θ_A, and Θ_H can be explicitly formed and checked for rank deficiencies without any information about the solution trajectories. If any of the matrices are unsatisfactorily conditioned, different parameter values must be chosen at which to sample the solution. Note that if $s < \max\{q_c, q_A, q_H\}$, then the problem is guaranteed to be ill-posed. Section 4 further addresses numerical issues that may arise from other sources of poor conditioning.

EXAMPLE 2.4 (Heat Equation). *Recall the problem introduced in Example 2.2. Given parameter samples $\{\mu_i = (\alpha_i, \beta_i)\}_{i=1}^s$, we learn a ROM of the form (2.9) by solving the pOpInf problem (2.13) with operator and data matrices*

$$\widehat{\mathbf{O}} = \begin{bmatrix} \widehat{\mathbf{A}}^{(1)} & \widehat{\mathbf{A}}^{(2)} \end{bmatrix} \in \mathbb{R}^{r\times 2r}, \qquad \mathbf{D} = \begin{bmatrix} \alpha_1\widehat{\mathbf{U}}(\mu_1)^\top & \beta_1\widehat{\mathbf{U}}(\mu_1)^\top \\ \vdots & \vdots \\ \alpha_s\widehat{\mathbf{U}}(\mu_s)^\top & \beta_s\widehat{\mathbf{U}}(\mu_s)^\top \end{bmatrix} \in \mathbb{R}^{sK\times 2r}.$$

Per Theorem 2.3, the rank of the data matrix $\mathbf{D}$ *depends on each* $\widehat{\mathbf{U}}(\mu_i)$ *and the matrix*

$$\Theta_A = \begin{bmatrix} \theta_A^{(1)}(\mu_1) & \theta_A^{(2)}(\mu_1) \\ \vdots & \vdots \\ \theta_A^{(1)}(\mu_s) & \theta_A^{(2)}(\mu_s) \end{bmatrix} = \begin{bmatrix} \alpha_1 & \beta_1 \\ \vdots & \vdots \\ \alpha_s & \beta_s \end{bmatrix} \in \mathbb{R}^{s\times 2}.$$

In particular, if Θ_A *does not have full column rank, then the data matrix* $\mathbf{D}$ *will be rank deficient. Our goal, then, is to choose* $s \geq 2$ *parameter samples such that* Θ_A *has a small condition number.*

Reduced-order models learned through pOpInf and those constructed via intrusive projection, i.e., by explicitly evaluating (2.6), are related in the following sense.

THEOREM 2.5. *Let* u *be the unique solution of (2.1a)–(2.1d), where the differential operator* $\mathcal{F}$ *has the quadratic, affine-parametric form described in (2.3a)–(2.3b). Suppose there exists an orthonormal set* $\{v_j\}_{j=1}^r \subset \mathcal{V}$*, and let* $\{\mu_i\}_{i=1}^s \subset \mathcal{P}$ *be a finite set of parameter samples. Define the loss function*

$$\mathscr{L}(\widehat{\mathbf{O}}) = \sum_{i=1}^{s} \int_{t_0}^{t_f} \left\| \mathbf{F}\left(\widehat{\mathbf{O}}; \hat{\mathbf{u}}, t, \theta, \mu_i\right) - \frac{\mathrm{d}}{\mathrm{d}t}\hat{\mathbf{u}}(t;\mu_i) \right\|_2^2 dt, \tag{2.15}$$

where $\hat{\mathbf{u}}(t;\mu) = [\langle v_1, u(\cdot,t;\mu)\rangle \ \cdots \ \langle v_r, u(\cdot,t;\mu)\rangle]^\top \in \mathbb{R}^r$*, and with* $\mathbf{F}$ *and* $\widehat{\mathbf{O}}$ *given by (2.10a)–(2.10b). Consider the following conditions.*

1. *There exist functions* $\hat{u}_j : [t_0,t_f]\times\mathcal{P}\to\mathbb{R}$*,* $j=1,\ldots,r$*, such that the finite sum representation (2.5) is exact for all* $x\in\Omega$*,* $t\in[t_0,t_f]$*, and* $\mu\in\{\mu_i\}_{i=1}^s$*, i.e., there is no truncation error at the sample parameter values.*
2. *For* $i=1,\ldots,s$*, there exist times* $\{\tau_{i,j}\}_{j=1}^K \subset [t_0,t_f]$*,* $K = 1+r+\binom{r+1}{2}$*, such that the matrix*

$$\widetilde{\mathbf{D}}(\mu_i) = \begin{bmatrix} \mathbf{1}_K & \widetilde{\mathbf{U}}(\mu_i)^\top & \left(\widetilde{\mathbf{U}}(\mu_i) \mathbin{\widehat{\odot}} \widetilde{\mathbf{U}}(\mu_i)\right)^\top \end{bmatrix} \in \mathbb{R}^{K\times K}$$

   is invertible, where $\widetilde{\mathbf{U}}(\mu_i) = [\hat{\mathbf{u}}(\tau_{i,1};\mu_i) \ \cdots \ \hat{\mathbf{u}}(\tau_{i,K};\mu_i)] \in \mathbb{R}^{r\times K}$*.*
3. *The matrices* $\Theta_c \in \mathbb{R}^{s\times q_c}$*,* $\Theta_A \in \mathbb{R}^{s\times q_A}$*, and* $\Theta_H \in \mathbb{R}^{s\times q_H}$ *of (2.14) have full column rank.*

If condition 1 holds, then the loss function $\mathscr{L}$ *has a global minimizer* $\widehat{\mathbf{O}}$ *satisfying* $\mathscr{L}(\widehat{\mathbf{O}}) = 0$*. If conditions 2 and 3 also hold, then that minimizer is unique.*

Proof. Assume condition 1 (no truncation error at the parameter samples). Then the system of ODEs (2.7a) with operators $\widehat{\mathbf{c}}^{(p)}$, $\widehat{\mathbf{A}}^{(p)}$, and $\widehat{\mathbf{H}}^{(p)}$ derived from (2.6) holds exactly for all $x\in\Omega$, $t\in[t_0,t_f]$, and $\mu\in\{\mu_i\}_{i=1}^s$, i.e., it is equivalent to (2.1a)–(2.1d) at the parameter samples. Constructing $\widehat{\mathbf{O}}$ from these operators, we have $\frac{\mathrm{d}}{\mathrm{d}t}\hat{\mathbf{u}}(t;\mu) = \mathbf{F}(\widehat{\mathbf{O}};\hat{\mathbf{u}},t,\theta,\mu)$ exactly for each $\mu\in\{\mu_i\}_{i=1}^s$. By construction, $\mathscr{L}(\widehat{\mathbf{O}}) = 0$, which—since $\mathscr{L}$ is non-negative—shows that $\widehat{\mathbf{O}}$ is a global minimizer of $\mathscr{L}$.

To prove uniqueness, assume conditions 2 and 3 and suppose $\widehat{\mathbf{O}}^{(1)}$ and $\widehat{\mathbf{O}}^{(2)}$ both minimize $\mathscr{L}$. By the previous argument, $\mathscr{L}(\widehat{\mathbf{O}}^{(1)}) = \mathscr{L}(\widehat{\mathbf{O}}^{(2)}) = 0$, which implies

$$\mathbf{0}_r = \mathbf{F}(\widehat{\mathbf{O}}^{(1)};\hat{\mathbf{u}},\tau_{i,j},\theta,\mu_i) - \mathbf{F}(\widehat{\mathbf{O}}^{(2)};\hat{\mathbf{u}},\tau_{i,j},\theta,\mu_i) = \mathbf{F}(\widetilde{\mathbf{O}};\hat{\mathbf{u}},\tau_{i,j},\theta,\mu_i) \tag{2.16}$$

for $i=1,\ldots,s$ and $j=1,\ldots,K$, where $\mathbf{0}_r\in\mathbb{R}^r$ is a column vector of zeros and

$$\widetilde{\mathbf{O}} = \widehat{\mathbf{O}}^{(1)} - \widehat{\mathbf{O}}^{(2)} = \left[\begin{array}{c|c|c} \widetilde{\mathbf{c}}^{(1)} \ \cdots \ \widetilde{\mathbf{c}}^{(q_c)} & \widetilde{\mathbf{A}}^{(1)} \ \cdots \ \widetilde{\mathbf{A}}^{(q_A)} & \widetilde{\mathbf{H}}^{(1)} \ \cdots \ \widetilde{\mathbf{H}}^{(q_H)} \end{array}\right] \in \mathbb{R}^{r\times q(r)}.$$

For fixed i, (2.16) with $j = 1, \ldots, K$ can be written as the linear system

$$\widetilde{\mathbf{D}}(\mu_i)\left[\ \sum_{p=1}^{q_c}\theta_c^{(p)}(\mu_i)\widetilde{\mathbf{c}}^{(p)}\ \middle|\ \sum_{p=1}^{q_A}\theta_A^{(p)}(\mu_i)\widetilde{\mathbf{A}}^{(p)}\ \middle|\ \sum_{p=1}^{q_H}\theta_H^{(p)}(\mu_i)\widetilde{\mathbf{H}}^{(p)}\ \right]^\top = \mathbf{0}_{K\times r},$$

where $\mathbf{0}_{K\times r} \in \mathbb{R}^{K\times r}$ is the zero matrix. As each $\widetilde{\mathbf{D}}(\mu_i)$ is invertible (condition 2), for $i = 1, \ldots, s$ we have

$$\sum_{p=1}^{q_c}\theta_c^{(p)}(\mu_i)\widetilde{\mathbf{c}}^{(p)} = \mathbf{0}_r, \quad \sum_{p=1}^{q_A}\theta_A^{(p)}(\mu_i)\widetilde{\mathbf{A}}^{(p)} = \mathbf{0}_{r\times r}, \quad \sum_{p=1}^{q_H}\theta_H^{(p)}(\mu_i)\widetilde{\mathbf{H}}^{(p)} = \mathbf{0}_{r\times\binom{r+1}{2}}.$$

Collecting these equations component-wise for $i = 1, \ldots, s$ yields the equations

$$\Theta_c\begin{bmatrix}[\widetilde{\mathbf{c}}^{(1)}]_j\\ \vdots \\ [\widetilde{\mathbf{c}}^{(q_c)}]_j\end{bmatrix} = \Theta_A\begin{bmatrix}[\widetilde{\mathbf{A}}^{(1)}]_{jk}\\ \vdots \\ [\widetilde{\mathbf{A}}^{(q_A)}]_{jk}\end{bmatrix} = \Theta_H\begin{bmatrix}[\widetilde{\mathbf{H}}^{(1)}]_{jk}\\ \vdots \\ [\widetilde{\mathbf{H}}^{(q_H)}]_{jk}\end{bmatrix} = \mathbf{0}_s$$

for every index pair j,k. But Θ_c, Θ_A, and Θ_H each have full column rank (condition 3), implying $[\widetilde{\mathbf{c}}^{(p)}]_j = [\widetilde{\mathbf{A}}^{(p)}]_{jk} = [\widetilde{\mathbf{H}}^{(p)}]_{jk} = 0$ for all j, k, p. Thus, $\widetilde{\mathbf{O}}$ is the zero matrix, hence $\widehat{\mathbf{O}}^{(1)} = \widehat{\mathbf{O}}^{(2)}$. ∎

Theorem 2.5 shows that for finite-dimensional PDEs (condition 1) with a polynomial, affine-parametric structure, the operators defined via (2.6) uniquely minimize the loss function $\mathscr{L}$ defined in (2.15), provided that the dynamics are sufficiently diverse (as quantified by conditions 2 and 3). As the time step between the data samples defining the pOpInf loss $\mathcal{L}$ of (2.11) decreases, i.e., $t_{j+1} - t_j \to 0$ for all j, then $\mathcal{L}$ converges to $\mathscr{L}$ (up to a constant), in which case the pOpInf ROM and the intrusive projection-based ROM will be the same. For infinite-dimensional PDEs, condition 1 is realized as the number of basis functions in the approximation increases, i.e., $r \to \infty$. Thus, there are two factors explaining why ROMs learned through pOpInf are not guaranteed to agree with (2.6): 1) the truncation error of the representation (2.5), and 2) the finite sampling of the solution in time. See [33, 46] for additional work connecting the intrusive and data-driven ROMs.

3. Affine Operator Inference for Systems of PDEs. Systems of partial differential equations with affine polynomial structure admit low-dimensional representations similar to (2.7a)–(2.7b) [17]. Consider the system of d partial differential equations where each equation can be written as in (2.1a)–(2.1d):

$$\frac{\partial u_\ell}{\partial t} = \mathcal{F}_\ell(u_\ell, \ldots, u_d; \mu), \qquad \ell = 1, \ldots, d, \tag{3.1a}$$

where each state variable u_ℓ is contained in a Hilbert space $\mathcal{V}_\ell$ of real-valued functions satisfying appropriate homogeneous boundary conditions. Suppose each spatial differential operator $\mathcal{F}_\ell : \mathcal{V}_1 \times \cdots \times \mathcal{V}_d \times \mathcal{P} \to \mathcal{V}_\ell^*$ is polynomial in the variables $u_1, \ldots, u_d$ and their spatial derivatives,

$$\mathcal{F}_\ell(u_1, \ldots, u_d; \mu) = \mathcal{C}_\ell(\mu) + \sum_{m=1}^{d}\mathcal{A}_{\ell,m}(u_m; \mu) + \sum_{m=1}^{d}\sum_{n=m}^{d}\mathcal{H}_{\ell,mn}(u_m, u_n; \mu), \tag{3.1b}$$

where $\mathcal{C}_\ell : \mathcal{P} \to \mathcal{V}_\ell^*$, $\mathcal{A}_{\ell,m} : \mathcal{V}_m \times \mathcal{P} \to \mathcal{V}_\ell^*$, $\mathcal{H}_{\ell,mn} : \mathcal{V}_m \times \mathcal{V}_n \times \mathcal{P} \to \mathcal{V}_\ell^*$, and each $\mathcal{A}_{\ell,m}$ and $\mathcal{H}_{\ell,mn}$ is linear in each state argument. This is the natural multivariate

extension of (2.3a). We further assume that each operator $\mathcal{C}_\ell$, $\mathcal{A}_{\ell,m}$, and $\mathcal{H}_{\ell,mn}$ has an affine-parametric expansion, that is,

$$\begin{aligned}\mathcal{C}_\ell(\mu) &= \sum_{p=1}^{q_{c_\ell}} \theta_{c_\ell}^{(p)}(\mu)\mathcal{C}_\ell^{(p)}. \qquad \mathcal{A}_{\ell,m}(u;\mu) = \sum_{p=1}^{q_{A_{\ell,m}}} \theta_{A_{\ell,m}}^{(p)}(\mu)\mathcal{A}_{\ell,m}^{(p)}(u).\\ \mathcal{H}_{\ell,mn}(u,v;\mu) &= \sum_{p=1}^{q_{H_{\ell,mn}}} \theta_{H_{\ell,mn}}^{(p)}(\mu)\mathcal{H}_{\ell,mn}^{(p)}(u,v).\end{aligned} \tag{3.1c}$$

For each $\ell = 1,\ldots,d$, let $\{v_{\ell j}\}_{j=1}^{r_\ell} \subset \mathcal{V}_\ell$ be an orthonormal set. We assume a reduced representation $\breve{u}_\ell$ of u_ℓ, confined to $\operatorname{span}(\{v_{\ell_1},\ldots,v_{\ell r_\ell}\}) \subset \mathcal{V}_\ell$, given by

$$\breve{u}_\ell(x,t;\mu) = \sum_{j=1}^{r_\ell} \hat{u}_{\ell j}(t;\mu)v_{\ell j}(x), \qquad \hat{u}_{\ell j}(t;\mu) = \langle v_{\ell j}, u_\ell(\cdot,t;\mu)\rangle_{\mathcal{V}_\ell}, \tag{3.2}$$

where $\langle\cdot,\cdot\rangle_{\mathcal{V}_\ell}$ is the duality pairing of $\mathcal{V}_\ell$ with its dual. Inserting $\breve{u}_\ell$ into (3.1a) for each ℓ and proceeding as before, we obtain d ODEs that serve as a ROM for (3.1a)–(3.1c):

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{u}}_\ell(t;\mu) &= \mathbf{F}_\ell(\widehat{\mathbf{O}}_\ell;\widehat{\mathbf{u}}_1,\ldots,\widehat{\mathbf{u}}_d,t,\theta,\mu)\\ &= \left(\sum_{p=1}^{q_{c_\ell}} \theta_{c_\ell}^{(p)}(\mu)\widehat{\mathbf{c}}_\ell^{(p)}\right) + \sum_{m=1}^{d}\left(\sum_{p=1}^{q_{A_{\ell,m}}} \theta_{A_{\ell,m}}^{(p)}(\mu)\widehat{\mathbf{A}}_{\ell,m}^{(p)}\right)\widehat{\mathbf{u}}_m(t;\mu)\\ &\quad + \sum_{m=1}^{d}\left(\sum_{p=1}^{q_{H_{\ell,mm}}} \theta_{H_{\ell,mm}}^{(p)}(\mu)\widehat{\mathbf{H}}_{\ell,mm}^{(p)}\right)\left(\widehat{\mathbf{u}}_m(t;\mu)\mathbin{\widehat{\odot}}\widehat{\mathbf{u}}_m(t;\mu)\right)\\ &\quad + \sum_{m=1}^{d}\sum_{n=m+1}^{d}\left(\sum_{p=1}^{q_{H_{\ell,mn}}} \theta_{H_{\ell,mn}}^{(p)}(\mu)\widehat{\mathbf{H}}_{\ell,mn}^{(p)}\right)\left(\widehat{\mathbf{u}}_m(t;\mu)\odot\widehat{\mathbf{u}}_n(t;\mu)\right),\end{aligned} \tag{3.3a}$$

$$\widehat{\mathbf{u}}_\ell(t;\mu) = \begin{bmatrix} \langle v_{\ell 1}, u_\ell(\cdot,t;\mu)\rangle & \cdots & \langle v_{\ell,r_\ell}, u_\ell(\cdot,t;\mu)\rangle \end{bmatrix}^\top \in \mathbb{R}^{r_\ell}, \tag{3.3b}$$

$$\widehat{\mathbf{O}}_\ell = \left[\begin{array}{ccc|ccc|ccc} \widehat{\mathbf{c}}_\ell^{(1)} & \cdots & \widehat{\mathbf{c}}_\ell^{(q_{c_\ell})} & \widehat{\mathbf{A}}_{\ell,1}^{(1)} & \cdots & \widehat{\mathbf{A}}_{\ell,d}^{(q_{A_{\ell,d}})} & \widehat{\mathbf{H}}_{\ell,11}^{(1)} & \cdots & \widehat{\mathbf{H}}_{\ell,dd}^{(q_{H_{\ell,dd}})} \end{array}\right], \tag{3.3c}$$

where $\widehat{\mathbf{c}}_\ell^{(p)} \in \mathbb{R}^{r_\ell}$, $\widehat{\mathbf{A}}_{\ell,m}^{(p)} \in \mathbb{R}^{r_\ell \times r_m}$, $\widehat{\mathbf{H}}_{\ell,mm}^{(p)} \in \mathbb{R}^{r_\ell \times \binom{r_m+1}{2}}$, and $\widehat{\mathbf{H}}_{\ell,mn}^{(p)} \in \mathbb{R}^{r_\ell \times r_m r_n}$ $(m \neq n)$. Here $\odot$ denotes the Khatri-Rao product, whereas $\mathbin{\widehat{\odot}}$ is the compact Khatri-Rao product that omits redundant terms (see Appendix A). Higher-order operators (e.g., cubic terms) can be similarly accounted for.

The low-dimensional system (3.3a) retains the differential and parametric structure of the PDE system (3.1a)–(3.1b). The corresponding affine pOpInf problem decouples into d instances of (2.13) which can be solved *independently*:

$$\min_{\widehat{\mathbf{O}}_\ell} \left\|\mathbf{D}_\ell\widehat{\mathbf{O}}_\ell^\top - \mathbf{R}_\ell^\top\right\|_F^2, \qquad \ell = 1,\ldots,d, \tag{3.4}$$

where

$$\mathbf{R}_\ell = \begin{bmatrix} \dot{\widehat{\mathbf{U}}}_\ell(\mu_1) & \cdots & \dot{\widehat{\mathbf{U}}}_\ell(\mu_s) \end{bmatrix} \in \mathbb{R}^{r_\ell \times sK},$$

$$\dot{\widehat{\mathbf{U}}}_\ell(\mu_i) = \left[\begin{array}{ccc} \frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{u}}_\ell(t;\mu_i)\Big|_{t=t_1} & \cdots & \frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{u}}_\ell(t;\mu_i)\Big|_{t=t_K} \end{array} \right] \in \mathbb{R}^{r_\ell \times K},$$

and where each $\mathbf{D}_\ell$ is constructed from the matrices $\widehat{\mathbf{U}}_1(\mu_i), \ldots, \widehat{\mathbf{U}}_d(\mu_i)$ where

$$\widehat{\mathbf{U}}_\ell(\mu_i) = \left[\begin{array}{ccc} \widehat{\mathbf{u}}_\ell(t_1;\mu_i) & \cdots & \widehat{\mathbf{u}}_\ell(t_K;\mu_i) \end{array} \right] \in \mathbb{R}^{r_\ell \times K}.$$

See Appendix C for the general construction, and note that the number of terms in the PDE dictates the number of terms in the ROM and hence the size of the pOpInf problem. This is best illustrated by example.

EXAMPLE 3.1 (FitzHugh-Nagumo System). *Let $\Omega = (0,1)$ and define the parameters $\mu = (\alpha, \beta, \gamma, \varepsilon) \in \mathbb{R}^4$. The following system of equations is a simplification of the Hodgkin-Huxley model for activation and deactivation in a spiking neuron [16, 30]:*

$$\frac{\partial u_1}{\partial t} = \varepsilon \frac{\partial^2 u_1}{\partial x^2} + \frac{1}{\varepsilon}\left(-u_1^3 + 1.1u_1^2 - 0.1u_1 - u_2 + \alpha\right), \tag{3.5a}$$

$$\frac{\partial u_2}{\partial t} = \beta u_1 - \gamma u_2 + \alpha, \tag{3.5b}$$

with initial conditions $u_1(x,t_0) = u_2(x,t_0) = 0$ and Neumann boundary conditions

$$\left.\frac{\partial u_1}{\partial x}\right|_{x=0} = f(t) := -50000t^3 e^{-15t}, \qquad \left.\frac{\partial u_1}{\partial x}\right|_{x=1} = 0. \tag{3.5c}$$

We write (3.5a)–(3.5c) in the language of (3.1a)–(3.1c) as

$$\begin{aligned}
\frac{\partial u_1}{\partial t} &= \theta^{(1)}_{c_1}(\mu)\,\mathcal{C}^{(1)}_1 + \theta^{(1)}_{A_{1,1}}(\mu)\,\mathcal{A}^{(1)}_{1,1}(u_1) + \theta^{(2)}_{A_{1,1}}(\mu)\,\mathcal{A}^{(2)}_{1,1}(u_1) + \theta^{(1)}_{A_{1,2}}(\mu)\,\mathcal{A}^{(2)}_{1,2}(u_2) \\
&\qquad + \theta^{(1)}_{H_{1,11}}(\mu)\,\mathcal{H}^{(1)}_{1,11}(u_1,u_1) + \theta^{(1)}_{G_{1,111}}(\mu)\,\mathcal{G}^{(1)}_{1,111}(u_1,u_1,u_1), \\
\frac{\partial u_2}{\partial t} &= \theta^{(1)}_{c_2}(\mu)\,\mathcal{C}^{(1)}_2 + \theta^{(1)}_{A_{2,1}}(\mu)\,\mathcal{A}^{(1)}_{2,1}(u_1) + \theta^{(1)}_{A_{2,2}}(\mu)\,\mathcal{A}^{(1)}_{2,2}(u_2),
\end{aligned}$$

with the operators and the affine coefficient functions given in Table 1. This motivates a ROM of the form

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{u}}_1(t;\mu) &= \theta^{(1)}_{c_1}(\mu)\,\widehat{\mathbf{c}}^{(1)}_1 + \left(\theta^{(1)}_{A_{1,1}}(\mu)\,\widehat{\mathbf{A}}^{(1)}_{1,1} + \theta^{(2)}_{A_{1,1}}(\mu)\,\widehat{\mathbf{A}}^{(2)}_{1,1}\right)\widehat{\mathbf{u}}_1(t;\mu) \\
&\qquad + \theta^{(1)}_{A_{1,2}}(\mu)\,\widehat{\mathbf{A}}^{(1)}_{1,2}\widehat{\mathbf{u}}_2(t;\mu) + \theta^{(1)}_{H_{1,11}}(\mu)\,\widehat{\mathbf{H}}^{(1)}_{1,11}\left(\widehat{\mathbf{u}}_1(t;\mu)\,\widehat{\odot}\,\widehat{\mathbf{u}}_1(t;\mu)\right) \\
&\qquad + \theta^{(1)}_{G_{1,111}}(\mu)\,\widehat{\mathbf{G}}^{(1)}_{1,111}\left(\widehat{\mathbf{u}}_1(t;\mu)\,\widehat{\odot}\,\widehat{\mathbf{u}}_1(t;\mu)\,\widehat{\odot}\,\widehat{\mathbf{u}}_1(t;\mu)\right) + \theta^{(1)}_{B_1}(\mu)\widehat{\mathbf{B}}^{(1)}_1 f(t), \\
\frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{u}}_2(t;\mu) &= \theta^{(1)}_{c_2}(\mu)\,\widehat{\mathbf{c}}^{(1)}_2 + \theta^{(1)}_{A_{2,1}}(\mu)\,\widehat{\mathbf{A}}^{(1)}_{2,1}\widehat{\mathbf{u}}_1(t;\mu) + \theta^{(1)}_{A_{2,2}}(\mu)\,\widehat{\mathbf{A}}^{(1)}_{2,2}\widehat{\mathbf{u}}_2(t;\mu),
\end{aligned}$$

where the sizes of each discretized operators are listed in Table 1. The operator $\widehat{\mathbf{B}}^{(1)}_1$ accounts for the Neumann boundary condition on u_1 in (3.5c); it has the coefficient function $\theta^{(1)}_{B_1}(\mu) = \varepsilon$. The term $\widehat{\mathbf{G}}^{(1)}_{1,111}$ represents the cubic nonlinearity for u_1.

The corresponding pOpInf problem is (3.4) with $d = 2$ and

$$\widehat{\mathbf{O}}_1 = \left[\begin{array}{ccccccc} \widehat{\mathbf{c}}^{(1)}_1 & \widehat{\mathbf{B}}^{(1)}_1 & \widehat{\mathbf{A}}^{(1)}_{1,1} & \widehat{\mathbf{A}}^{(2)}_{1,1} & \widehat{\mathbf{A}}^{(1)}_{1,2} & \widehat{\mathbf{H}}^{(1)}_{1,11} & \widehat{\mathbf{G}}^{(1)}_{1,111} \end{array} \right] \in \mathbb{R}^{r_1 \times q_1(r_1,r_2)},$$

$$\mathbf{D}_1 = \left[\begin{array}{c|c|c|c} \mathbf{D}_{c_1} & \mathbf{D}_{A_1} & \mathbf{D}_{H_1} & \mathbf{D}_{G_1} \end{array} \right] \in \mathbb{R}^{sK \times q_1(r_1,r_2)},$$

$$\mathbf{D}_{c_1} = \begin{bmatrix} \theta^{(1)}_{c_1}(\mu_1)\mathbf{1}_K & \theta^{(1)}_{B_1}(\mu_1)\mathbf{f} \\ \vdots & \vdots \\ \theta^{(1)}_{c_1}(\mu_s)\mathbf{1}_K & \theta^{(1)}_{B_1}(\mu_s)\mathbf{f} \end{bmatrix},$$

$$\mathbf{D}_{A_1} = \begin{bmatrix} \theta^{(1)}_{A_{1,1}}(\mu_1)\widehat{\mathbf{U}}_1(\mu_1)^\top & \theta^{(2)}_{A_{1,1}}(\mu_1)\widehat{\mathbf{U}}_1(\mu_1)^\top & \theta^{(1)}_{A_{1,2}}(\mu_1)\widehat{\mathbf{U}}_2(\mu_1)^\top \\ \vdots & \vdots & \vdots \\ \theta^{(1)}_{A_{1,1}}(\mu_s)\widehat{\mathbf{U}}_1(\mu_s)^\top & \theta^{(2)}_{A_{1,1}}(\mu_s)\widehat{\mathbf{U}}_1(\mu_s)^\top & \theta^{(1)}_{A_{1,2}}(\mu_s)\widehat{\mathbf{U}}_2(\mu_s)^\top \end{bmatrix},$$

$$\mathbf{D}_{H_1} = \begin{bmatrix} \theta^{(1)}_{H_{1,11}}(\mu_1)\left(\widehat{\mathbf{U}}_1(\mu_1)\,\widehat{\odot}\,\widehat{\mathbf{U}}_1(\mu_1)\right)^\top \\ \vdots \\ \theta^{(1)}_{H_{1,11}}(\mu_s)\left(\widehat{\mathbf{U}}_1(\mu_s)\,\widehat{\odot}\,\widehat{\mathbf{U}}_1(\mu_s)\right)^\top \end{bmatrix},$$

$$\mathbf{D}_{G_1} = \begin{bmatrix} \theta^{(1)}_{G_{1,111}}(\mu_1)\left(\widehat{\mathbf{U}}_1(\mu_1)\,\widehat{\odot}\,\widehat{\mathbf{U}}_1(\mu_1)\,\widehat{\odot}\,\widehat{\mathbf{U}}_1(\mu_1)\right)^\top \\ \vdots \\ \theta^{(1)}_{G_{1,111}}(\mu_s)\left(\widehat{\mathbf{U}}_1(\mu_s)\,\widehat{\odot}\,\widehat{\mathbf{U}}_1(\mu_s)\,\widehat{\odot}\,\widehat{\mathbf{U}}_1(\mu_s)\right)^\top \end{bmatrix},$$

$$\widehat{\mathbf{O}}_2 = \begin{bmatrix} \widehat{\mathbf{c}}^{(1)}_2 & \widehat{\mathbf{A}}^{(1)}_{2,1} & \widehat{\mathbf{A}}^{(1)}_{2,2} \end{bmatrix} \in \mathbb{R}^{r_2 \times q_2(r_1,r_2)},$$

$$\mathbf{D}_2 = \begin{bmatrix} \theta^{(1)}_{c_2}(\mu_1)\mathbf{1}_K & \theta^{(1)}_{A_{2,1}}(\mu_1)\widehat{\mathbf{U}}_1(\mu_1)^\top & \theta^{(1)}_{A_{2,2}}(\mu_1)\widehat{\mathbf{U}}_2(\mu_1)^\top \\ \vdots & \vdots & \vdots \\ \theta^{(1)}_{c_2}(\mu_s)\mathbf{1}_K & \theta^{(1)}_{A_{2,1}}(\mu_s)\widehat{\mathbf{U}}_1(\mu_s)^\top & \theta^{(1)}_{A_{2,2}}(\mu_s)\widehat{\mathbf{U}}_2(\mu_s)^\top \end{bmatrix} \in \mathbb{R}^{sK \times q_2(r_1,r_2)},$$

where $\mathbf{f} = [f(t_1) \ \cdots \ f(t_K)]^\top \in \mathbb{R}^K$, $q_1(r_1,r_2) = 2 + 2r_1 + r_2 + \binom{r_1+1}{2} + \binom{r_1+2}{3}$, *and* $q_2(r_1,r_2) = 1 + r_1 + r_2$. *The data matrix* $\mathbf{D}_1$ *and the operator matrix* $\widehat{\mathbf{O}}_1$ *have been modified from* (3.3a)–(3.3c) *to account for the cubic term present* (3.5a).

REMARK 3.2. *Even though since [15], the FitzHugh-Nagumo system has been widely used as a benchmark problem for the development of general nonlinear model reduction methods, a fully cubic intrusive model can be directly derived for this system (i.e., the nonlinear terms in the system are point-wise local and exactly quadratic and cubic), circumventing the need for the second layer of approximation introduced through hyper-reduction.*

4. Computational Procedure. Solving the pOpInf problem (2.13) requires samples of the solution $u(x,t;\mu)$ and its time derivative at times $\{t_j\}_{j=1}^K$ for each selected parameter value $\{\mu_i\}_{i=1}^s$. The quality of the resulting ROM depends on how well the orthonormal basis functions $\{v_j\}_{j=1}^r$ represent the solution at each parameter value and throughout the spatial and temporal domains. We therefore adopt the widely used proper orthogonal decomposition (POD) [1, 10, 28, 41], defined by the set of orthonormal functions that minimize the mean squared projection error of the sample data, i.e., solving the problem

$$\min_{v_1,\ldots,v_r \in \mathcal{V}} \sum_{i=1}^{s}\sum_{j=1}^{K} \left\| u(\cdot,t_j;\mu_i) - \sum_{\ell=1}^{r} \langle v_\ell, u(\cdot,t_j;\mu_i)\rangle v_\ell \right\|_{\mathcal{V}}^2 \quad \text{subject to} \quad \langle v_i, v_j\rangle = \delta_{ij},$$

TABLE 1

The operators in the continuous setting, the associated affine coefficient functions, and the size of the reduced operators in discretized setting for the FitzHugh-Nagumo system of Example 3.1.

		continuous term	affine coefficient	discretized operator
Eq. (3.5a)	constant	$\mathcal{C}_1^{(1)} = 1$	$\theta_{c_1}^{(1)}(\mu) = \alpha/\varepsilon$	$\widehat{\mathbf{c}}_1^{(1)} \in \mathbb{R}^{r_1}$
	input	$\frac{\partial u_1}{\partial x}\big\|_{x=0} = f(t)$	$\theta_{B_1}^{(1)}(\mu) = \varepsilon$	$\widehat{\mathbf{B}}_1^{(1)} \in \mathbb{R}^{r_1}$
	linear	$\mathcal{A}_{1,1}^{(1)}(u) = \frac{\partial^2 u}{\partial x^2}$	$\theta_{A_{1,1}}^{(1)}(\mu) = \varepsilon$	$\widehat{\mathbf{A}}_{1,1}^{(1)} \in \mathbb{R}^{r_1 \times r_1}$
		$\mathcal{A}_{1,1}^{(2)}(u) = u$	$\theta_{A_{1,1}}^{(2)}(\mu) = -0.1/\varepsilon$	$\widehat{\mathbf{A}}_{1,1}^{(2)} \in \mathbb{R}^{r_1 \times r_1}$
		$\mathcal{A}_{1,2}^{(1)}(u) = u$	$\theta_{A_{1,2}}^{(1)}(\mu) = -1/\varepsilon$	$\widehat{\mathbf{A}}_{1,2}^{(1)} \in \mathbb{R}^{r_1 \times r_2}$
	quadratic	$\mathcal{H}_{1,11}^{(1)}(u,v) = uv$	$\theta_{H_{1,11}}^{(1)}(\mu) = 1.1/\varepsilon$	$\widehat{\mathbf{H}}_{1,11}^{(1)} \in \mathbb{R}^{r_1 \times \binom{r_1+1}{2}}$
	cubic	$\mathcal{G}_{1,111}^{(1)}(u,v,w) = uvw$	$\theta_{G_{1,111}}^{(1)}(\mu) = -1/\varepsilon$	$\widehat{\mathbf{G}}_{1,111}^{(1)} \in \mathbb{R}^{r_1 \times \binom{r_1+2}{3}}$
Eq. (3.5b)	constant	$\mathcal{C}_2^{(1)} = 1$	$\theta_{c_2}^{(1)}(\mu) = \alpha$	$\widehat{\mathbf{c}}_2^{(1)} \in \mathbb{R}^{r_2}$
	linear	$\mathcal{A}_{2,1}^{(1)}(u) = u$	$\theta_{A_{2,1}}^{(1)}(\mu) = \beta$	$\widehat{\mathbf{A}}_{2,1}^{(1)} \in \mathbb{R}^{r_2 \times r_1}$
		$\mathcal{A}_{2,2}^{(1)}(u) = u$	$\theta_{A_{2,2}}^{(1)}(\mu) = -\gamma$	$\widehat{\mathbf{A}}_{2,2}^{(1)} \in \mathbb{R}^{r_2 \times r_2}$

where $\|v\|_{\mathcal{V}} = \sqrt{\langle v, v \rangle}$ is the natural norm on $\mathcal{V}$. This data-driven choice of basis optimally represents the solution at the sampled parameter values, although the resulting ROM does not share such guarantees [9].

The conditioning of the linear least-squares problem (2.13) depends on the data matrix $\mathbf{D}$. If the parameter samples are chosen so that the pitfalls described in Theorem 2.3 are avoided, then the condition number of $\mathbf{D}$ depends on the nature of the solution at the quadrature points $\{t_j\}_{j=1}^K$. To improve the conditioning, we introduce a Tikhonov regularization [44] so that (2.13) becomes

$$\min_{\widehat{\mathbf{O}}} \left\| \mathbf{D}\widehat{\mathbf{O}}^\top - \mathbf{R}^\top \right\|_F^2 + \left\| \mathbf{\Lambda}\widehat{\mathbf{O}}^\top \right\|_F^2, \qquad \mathbf{\Lambda} \in \mathbb{R}^{q(r) \times q(r)}. \tag{4.1}$$

The solution to this regularized problem satisfies the modified normal equations,

$$\left(\mathbf{D}^\top \mathbf{D} + \mathbf{\Lambda}^\top \mathbf{\Lambda} \right) \widehat{\mathbf{O}}^\top = \mathbf{D}^\top \mathbf{R}^\top. \tag{4.2}$$

The regularizer $\mathbf{\Lambda}$ can be parameterized in a number of ways [29]. One choice that provides flexibility without introducing a large number of hyperparameters is the diagonal matrix $\mathbf{\Lambda} = \mathbf{\Lambda}(\lambda_1, \lambda_2)$, defined such that

$$\left\| \mathbf{\Lambda}(\lambda_1, \lambda_2)\widehat{\mathbf{O}}^\top \right\|_F^2 = \lambda_1^2 \left(\sum_{p=1}^{q_c} \left\| \widehat{\mathbf{c}}^{(p)} \right\|_2^2 + \sum_{p=1}^{q_A} \left\| \widehat{\mathbf{A}}^{(p)} \right\|_F^2 \right) + \lambda_2^2 \sum_{p=1}^{q_H} \left\| \widehat{\mathbf{H}}^{(p)} \right\|_F^2. \tag{4.3}$$

This regularization structure groups the operators defining the ROM according to their polynomial order and drives $\widehat{\mathbf{O}}$ toward the zero matrix as $\lambda_1, \lambda_2 \to \infty$. Therefore, the regularization drives the resulting ROM toward the globally stable zero system $\frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{u}}(t;\mu) = \mathbf{0}$. As before, the extension of (4.1) to a system of PDEs is straightforward where each PDE is written as an independent pOpInf problem (see (3.4)).

We choose the regularization hyperparameters $\lambda_1, \lambda_2 \geq 0$ to minimize the mean

squared training error

$$\frac{1}{s}\sum_{i=1}^{s}\sum_{j=1}^{K}\|\widehat{\mathbf{u}}(t_j;\mu_i)-\widetilde{\mathbf{u}}(t_j;\mu_i)\|_2^2,$$

where $\widehat{\mathbf{u}}(t_j;\mu_i)$ is the training data and $\widetilde{\mathbf{u}}(t_j;\mu_i)$ is the result of integrating the ODE $\frac{\mathrm{d}}{\mathrm{d}t}\widetilde{\mathbf{u}}(t)=\mathbf{F}(\widehat{\mathbf{O}};\widetilde{\mathbf{u}},t,\mu)$ defined by the solution $\widehat{\mathbf{O}}$ of (4.1) with regularization hyperparameters λ_1 and λ_2. This is an optimization problem in the principal quadrant of $\mathbb{R}^2$ (but is otherwise free of constraints), which we carry out with a sparse grid search followed by a derivative-free search method [31].

We now summarize the computational procedure for solving (2.13): 1) select parameter values $\{\mu_i\}_{i=1}^s$ such that the associated $\Theta_c,\ldots,\Theta_G$ have full column rank; 2) sample the PDE solution $u(x,t,\mu)$ and its time derivative for $t\in\{t_j\}_{j=1}^K$ for each $\mu\in\{\mu_i\}_{i=1}^s$; 3) compute the POD basis associated with the sampled solution data; 4) use the POD basis to project the solution data, obtaining $\widehat{\mathbf{U}}(\mu_i)$ and $\dot{\widehat{\mathbf{U}}}(\mu_i)$; 5) form the data matrix $\mathbf{D}$ and the time derivative matrix $\mathbf{R}$; 6) choose optimal regularization hyperparameters and solve (4.1) with these hyperparameters. In the computational setting, we sample the solution in step 2 by obtaining approximate discretized *solution snapshots* via a high-fidelity solver. For example, let $\{\mathbf{x}_\ell\}_{\ell=1}^N\subset\Omega$ be a discretization of Ω, and define

$$\mathbf{U}(\mu_i)=\begin{bmatrix} u(\mathbf{x}_1,t_1;\mu_i) & \cdots & u(\mathbf{x}_1,t_K;\mu_i)\\ \vdots & & \vdots\\ u(\mathbf{x}_N,t_1;\mu_i) & \cdots & u(\mathbf{x}_N,t_K;\mu_i)\end{bmatrix}\in\mathbb{R}^{N\times K},$$

the *snapshot matrix* for parameter μ_i. The rank-r POD basis of step 3 is comprised of the first r left singular vectors of the concatenated snapshot matrices, that is,

$$\mathbf{\Phi}\mathbf{\Sigma}\mathbf{\Psi}^\top=\begin{bmatrix}\mathbf{U}(\mu_1) & \cdots & \mathbf{U}(\mu_s)\end{bmatrix}\in\mathbb{R}^{N\times sK},\qquad \mathbf{V}=\mathbf{\Phi}_{:,1:r}\in\mathbb{R}^{N\times r},\tag{4.4}$$

where $\mathbf{\Phi}\mathbf{\Sigma}\mathbf{\Psi}^\top$ is the singular value decomposition (SVD). With this notation, the projection of step 4 is given by $\widehat{\mathbf{U}}(\mu_i)=\mathbf{V}^\top\mathbf{U}(\mu_i)$, $i=1,\ldots,s$. If the time derivatives of u are not provided by the high-fidelity solver, they may be estimated as finite differences of the solution snapshots. The time integration error of the high-fidelity model, as well as the approximation error accompanying finite differences for the time derivatives, are additional motivations for utilizing the regularization strategy described previously. Algorithm 4.1 fully details the procedure. The algorithm is presented for the general case of a system of d partial differential equations, but it may be simplified to a case of a single PDE by setting $d=1$.

5. Numerical Examples. We now present numerical results for the heat equation and FitzHugh-Nagumo system introduced in Section 2 and Section 3, respectively.

5.1. Heat Equation. We return to the heat equation of Example 2.2 and 2.4, setting $\bar{x}=2/3$ and $u_0(x;\mu)=1-(1-x)^{50}-x^{50}$. For each of the $s=5$ parameter samples shown in Figure 1, we generate high-fidelity solutions by discretizing the spatial domain $\Omega=(0,1)$ with a uniform grid of $N=1000$ points and approximating

Algorithm 4.1 Regularized **p**arametric **Op**erator **Inf**erence for systems of PDEs

procedure pOpInf(training parameters $\mu_1, \ldots, \mu_s \in \mathcal{P}$, high-fidelity training snapshots $\mathbf{U}_1(\mu_i), \ldots, \mathbf{U}_d(\mu_i) \in \mathbb{R}^{N \times K}$ for $i = 1, \ldots, s$, affine coefficient functions $\theta = \{\theta_{c_1}^{(1)}, \ldots, \theta_{H_{d,dd}}^{(q_{H_{d,dd}})}\} : \mathcal{P} \to \mathbb{R}$, reduced dimensions $r_1, \ldots, r_d \in \mathbb{N}$)

  `# Project training data to low-dimensional subspaces.`
  for $\ell = 1, \ldots, d$ **do**
    $\mathbf{V}_\ell \leftarrow \texttt{pod}([\mathbf{U}_\ell(\mu_1) \ \cdots \ \mathbf{U}_\ell(\mu_s)], r_\ell)$ `# Rank-`r_ℓ` POD basis.`
    for $i = 1, \ldots, s$ **do**
      $\widehat{\mathbf{U}}_\ell(\mu_i) \leftarrow \mathbf{V}_\ell^\top \mathbf{U}_\ell(\mu_i)$ `# Projected solution data.`
      $\dot{\widehat{\mathbf{U}}}_\ell(\mu_i) \leftarrow \frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{U}}_\ell(\mu_i)$ `# Projected time derivatives.`
  `# Construct pOpInf matrices.`
  for $\ell = 1, \ldots, d$ **do**
    $\mathbf{D}_\ell \leftarrow$ build the ℓth data matrix from $\widehat{\mathbf{U}}_1(\mu_1), \ldots, \widehat{\mathbf{U}}_d(\mu_s), \theta, \mu_1, \ldots, \mu_s$
    $\mathbf{R}_\ell \leftarrow [\dot{\widehat{\mathbf{U}}}_\ell(\mu_1) \ \cdots \ \dot{\widehat{\mathbf{U}}}_\ell(\mu_s)]$
  `# Compute pOpInf solution with optimal hyperparameters.`
  $\lambda_1^*, \lambda_2^* \leftarrow \operatorname{argmin}$ TrainingError(λ_1, λ_2)
  return RegOpInf$(\lambda_1^*, \lambda_2^*)$

procedure TrainingError(λ_1, λ_2)
  $\widehat{\mathbf{O}}_1, \ldots, \widehat{\mathbf{O}}_d \leftarrow$ RegOpInf(λ_1, λ_2)
  for $i = 1, \ldots, s$ **do** `# Calculate error at training parameters.`
    $\widetilde{\mathbf{U}}_1(\mu_i), \ldots, \widetilde{\mathbf{U}}_d(\mu_i) \leftarrow$ integrate (3.3a), $\ell = 1, \ldots, d$, over $[t_0, t_f]$
  return $\frac{1}{sd}\sum_{\ell=1}^{d}\sum_{i=1}^{s} \|\widehat{\mathbf{U}}_\ell(\mu_i) - \widetilde{\mathbf{U}}_\ell(\mu_i)\|_F^2$

procedure RegOpInf(λ_1, λ_2)
  $\mathbf{\Lambda}^2 \leftarrow \mathbf{\Lambda}(\lambda_1, \lambda_2)^\top \mathbf{\Lambda}(\lambda_1, \lambda_2)$ `# Construct the regularizer.`
  for $\ell = 1, \ldots, d$ **do**
    $\widehat{\mathbf{O}}_\ell^\top \leftarrow \left(\mathbf{D}_\ell^\top \mathbf{D}_\ell + \mathbf{\Lambda}^2\right)^{-1} \mathbf{D}_\ell^\top \mathbf{R}_\ell^\top$ `# Solve the` ℓ`th pOpInf problem.`
  return $\widehat{\mathbf{O}}_1, \ldots, \widehat{\mathbf{O}}_d$

the spatial derivative with second-order central finite differences, then integrating the resulting semi-discrete ODE in time with the first-order implicit Euler scheme on $K = 1500$ uniformly spaced time steps in $[t_0, t_f] = [0, 1.5]$. Figure 1 also shows example snapshots for each parameter sample, demonstrating that variation in the parameters $\mu = (\alpha, \beta)$ determines the diffusion dynamics. To select the number of modes in the POD basis, define the cumulative energy

$$\mathcal{E}(r) = \sum_{j=1}^{r} \sigma_j^2 \Big/ \sum_{j=1}^{N} \sigma_j^2,$$

where σ_j is the jth singular value in the POD factorization (4.4). Note that $\mathcal{E}(r)$ is a nondecreasing function of r. We select r to be the smallest integer such that the residual energy $1 - \mathcal{E}(r)$, or the energy in the non-retained modes, lies below a fixed threshold $\epsilon > 0$ (see Figure 2). In this case, setting $\epsilon = 10^{-12}$ results in $r = 24$.

We compute $\widehat{\mathbf{O}}$ using Algorithm 4.1. Since (2.9) is a linear system, the regular-

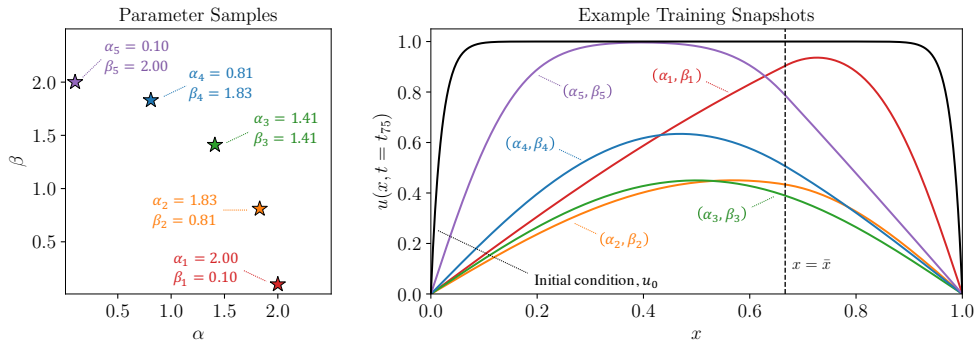


FIG. 1. *Experimental parameter samples (left) and associated snapshots at intermediate time* $t = t_{75} = 0.075$ *(right) for the heat equation problem (2.8a)–(2.8c). The vertical line* $x = \bar{x}$ *marks the point in the domain where the diffusion constant switches between* α *and* β.

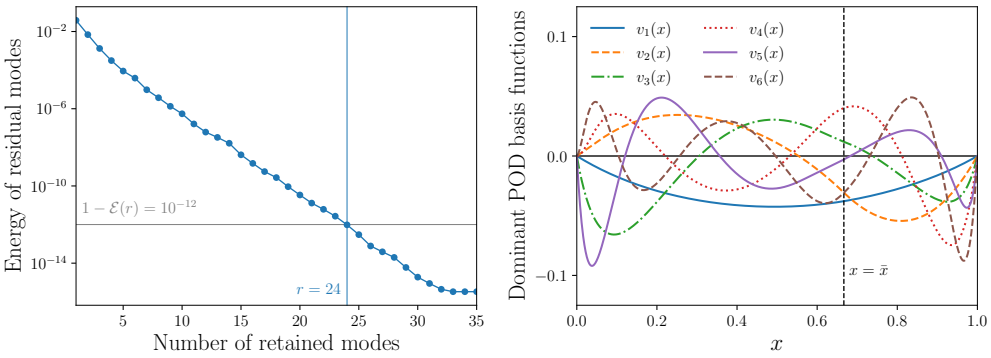


FIG. 2. *Residual energy decay (left) and the six dominant POD basis functions (right) for the snapshot set generated at the parameter samples in Figure 1. For* $r \geq 24$, *we have* $1 - \mathcal{E}(r) < 10^{-12}$.

ization is parameterized by a single hyperparameter, i.e., we minimize the residual

$$\min_{\widehat{\mathbf{O}}} \left\| \mathbf{D}\widehat{\mathbf{O}}^\top - \mathbf{R}^\top \right\|_F^2 + \lambda_1^2 \left(\|\widehat{\mathbf{A}}^{(1)}\|_F^2 + \|\widehat{\mathbf{A}}^{(2)}\|_F^2 \right).$$

To evaluate the performance of the resulting ROM in terms of the parameters, we discretize $\mathcal{P}$ in a 40×40 uniform grid and, for each μ in the grid, compute high-fidelity solutions $\mathbf{u}(t_1;\mu), \ldots, \mathbf{u}(t_K;\mu) \in \mathbb{R}^N$ and integrate the ROM to obtain reduced states $\widetilde{\mathbf{u}}(t_1;\mu), \ldots, \widetilde{\mathbf{u}}(t_K;\mu) \in \mathbb{R}^r$, then compute the relative L_2-norm error

$$\frac{\|\mathbf{V}\widetilde{\mathbf{u}}(\cdot;\mu) - \mathbf{u}(\cdot;\mu)\|_{L^2([t_0,t_f])}}{\|\mathbf{u}(\cdot;\mu)\|_{L^2([t_0,t_f])}}, \quad \|\mathbf{w}(\cdot)\|_{L^2([t_0,t_f])} = \left(\int_{t_0}^{t_f} \|\mathbf{w}(t)\|_2^2 \, dt \right)^{1/2}, \tag{5.1}$$

estimating the time integrals via the trapezoidal rule. Figure 3 shows the results and compares them to the relative projection error induced by the basis, given by

$$\frac{\left\|\mathbf{u}(\cdot;\mu) - \mathbf{V}\mathbf{V}^\top \mathbf{u}(\cdot;\mu)\right\|_{L^2([t_0,t_f])}}{\|\mathbf{u}(\cdot;\mu)\|_{L^2([t_0,t_f])}}. \tag{5.2}$$

The relative projection error (5.2) is on the order of 10^{-7} throughout $\mathcal{P}$, which indicates that the training snapshots have sufficient information to represent the solution

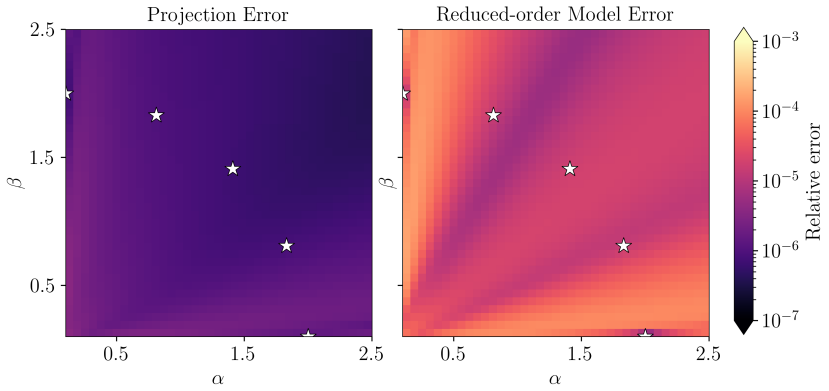


FIG. 3. *Relative L^2 projection errors (left) and pOpInf ROM errors (right) over the parameter domain $\mathcal{P}$ for the problem (2.8a)–(2.8c). The parameter samples used to generate the training set are marked as stars and $r = 24$ basis functions are used. The geometric means of the projection and ROM errors are approximately 7.68×10^{-7} and 3.48×10^{-5}, respectively.*

well for any $\mu \in \mathcal{P}$. The ROM error (5.1) is on the order of 10^{-5} throughout $\mathcal{P}$ (less than 0.02% everywhere) and runs with a computational speedup of ~ 10 times compared to the high-fidelity model. The results in Figure 3 highlight the ability of the ROM to generalize beyond the training data: in particular, the ROM performs well for (α, β) pairs away from the arc $\alpha^2 + \beta^2 = 4$ where the parameter samples lie.

5.2. FitzHugh-Nagumo System. The neuron model (3.5a)–(3.5b) introduced in Example 3.1 features a four-dimensional parameter space. We generate training data at each of the $504 = 6 \times 6 \times 2 \times 7$ unique parameters $\mu = (\alpha, \beta, \gamma, \varepsilon)$ for

$$\text{(training set)} \qquad \begin{aligned} &\alpha \in \{0.025, 0.035, \ldots, 0.075\}, \quad \beta \in \{0.25, 0.35, \ldots, 0.75\}, \\ &\gamma \in \{2.0, 2.5\}, \qquad\qquad\qquad \varepsilon \in \{0.010, 0.015, \ldots, 0.040\}. \end{aligned}$$

The full-order problem is solved by discretizing the domain $\Omega = (0, 1)$ with $N_x = 512$ spatial points (so the total spatial dimension is $N = 2N_x = 1024$), approximating the differential term with central finite differences, and integrating the equation in time via the first-order implicit-explicit Euler scheme (treating the linear terms with a backward Euler step and the nonlinear terms with a forward Euler step [5]) with step size $\delta t = 10^{-3}$ over $[t_0, t_f] = [0, 4]$. Every 10th snapshot is recorded, resulting in $K = 400$ training snapshots per parameter sample. Figure 4 shows the phase plot of u_1 against u_2 at multiple spatial coordinate values for three of the training parameter samples, demonstrating that the system exhibits a diverse range of dynamical behaviors as the parameter values are varied. To evaluate ROM performance with respect to the parameters, we also solve the full-order model at the $10{,}749 = 11 \times 11 \times 3 \times 31 - 504$ additional parameters $\mu = (\alpha, \beta, \gamma, \varepsilon)$ with

$$\text{(testing set)} \qquad \begin{aligned} &\alpha \in \{0.025, 0.030, \ldots, 0.075\}, \quad \beta \in \{0.25, 0.30, \ldots, 0.75\}, \\ &\gamma \in \{2.00, 2.25, 2.50\}, \qquad\qquad \varepsilon \in \{0.010, 0.011, \ldots, 0.040\}. \end{aligned}$$

Parameters from the training set are not included in the testing set.

The FitzHugh-Nagumo system (3.5a)–(3.5b) exhibits a rapid singular Hopf bifurcation with respect to ε wherein the limit cycle collapses to a stable fixed point as ε increases [6]. At parameters near this transition, the full-order model and ROM solutions are highly sensitive to ε and the time step δt. Including such parameters

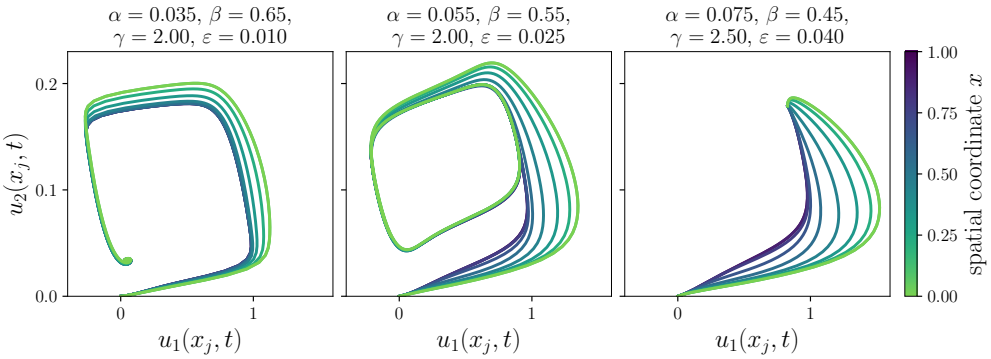
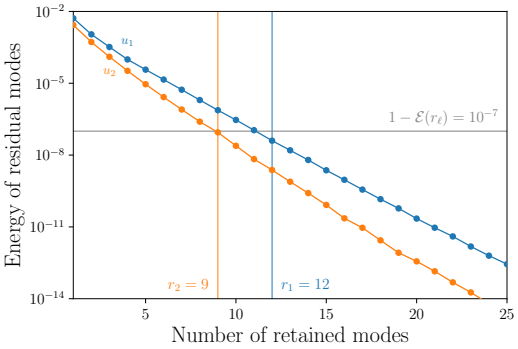


FIG. 4. *Phase portraits of training trajectories for the FitzHugh-Nagumo system (3.5a)–(3.5b), traced out at various points in the spatial domain. The center trajectory has a limit cycle, while the trajectories on the left and right converge to a single point. Small ε values de-emphasize the diffusion term and drive the system toward spatial homogeneity; larger ε values result in more variation across the spatial domain but which decreases with time.*



$1 - \mathcal{E}(r_\ell)$	r_1	r_2
10^{-3}	3	2
10^{-4}	4	4
10^{-5}	7	5
10^{-6}	9	7
10^{-7}	12	9
10^{-8}	14	11
10^{-9}	16	13
10^{-10}	19	15
10^{-11}	21	17
10^{-12}	24	19

FIG. 5. *Decay of the residual energy in the training data for u_1 and u_2 in the FitzHugh-Nagumo problem (3.5a)–(3.5b) (left) and the corresponding selected basis sizes (right). Demanding that $1 - \mathcal{E}(r_\ell) < 10^{-7}$ requires $r_1 = 12$ and $r_2 = 9$ POD modes for u_1 and u_2, respectively.*

in the training and testing sets makes it difficult to assess ROM accuracy. We select $\delta t = 10^{-3}$ as the baseline time step for which most of the full-order simulations converge, then eliminate the small number of points from our training and testing sets for which refining the time step from $\delta t = 10^{-3}$ to $\delta t = 10^{-5}$ results in a 30% change in the full-order solution as measured by the $L^2([t_0, t_f])$ norm. This prompts us to remove five training parameters ($\sim 1\%$ of the original training set) and fifty-two testing parameters ($\sim 0.5\%$ of the original testing set). Hence the size of the final training set is $s = 504 - 5 = 499$ and the size of the final testing set is $10{,}749 - 52 = 10{,}697$, with size ratio $10{,}697/499 \approx 21.4$. With this approach, ROMs obtained through pOpInf are stable throughout almost the entire testing set; for the few testing parameters where the ROMs are unstable with time step $\delta t = 10^{-3}$, the time step is adjusted to maintain stability.

We select the number of POD modes r_1 and r_2 for the variables u_1 and u_2, respectively, based on the residual energy $1 - \mathcal{E}(r_\ell)$ of the training data, $\ell = 1, 2$. For each of the resulting (r_1, r_2) pairs shown in Figure 5, we learn a pOpInf ROM via Algorithm 4.1. To simplify the hyperparameter search, we regularize only the

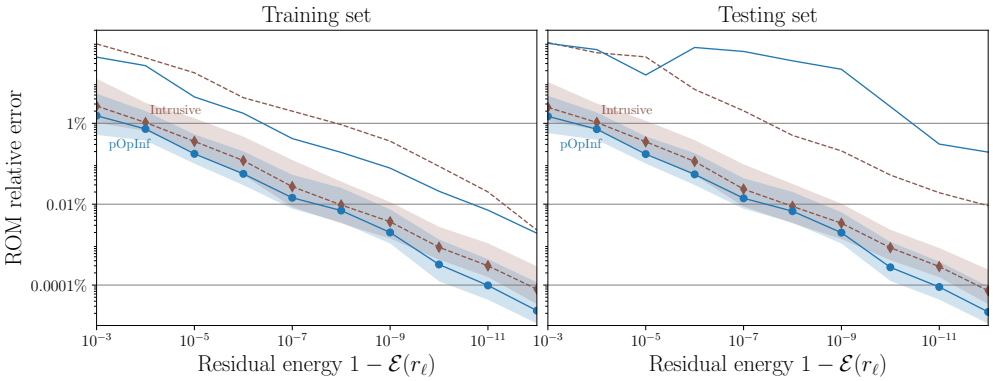


FIG. 6. *Relative errors (calculated over space and time) of the learned pOpInf ROM and the intrusive ROM for the FitzHugh-Nagumo system. As residual energy decreases from left to right, the sizes of the underlying POD bases increase according to Figure 5. The shaded regions show the* 10%–90% *interdecile range of the error across all training samples (left) or testing samples (right), with the corresponding median and maximum errors denoted by the lines for the pOpInf ROMs and dashed lines for the intrusive ROMs.*

quadratic and cubic terms, so that the regularized pOpInf data residuals are

$$\min_{\widehat{\mathbf{O}}_1} \left\| \mathbf{D}_1 \widehat{\mathbf{O}}_1^\top - \mathbf{R}_1^\top \right\|_F^2 + \lambda_2 \left\| \widehat{\mathbf{H}}_{1,11}^{(1)} \right\|_F^2 + \lambda_3 \left\| \widehat{\mathbf{G}}_{1,111}^{(1)} \right\|_F^2, \quad \min_{\widehat{\mathbf{O}}_2} \left\| \mathbf{D}_2 \widehat{\mathbf{O}}_2^\top - \mathbf{R}_2^\top \right\|_F^2.$$

Regularization of the additional terms (λ_1 in (4.3)) was observed to have a marginal effect on the results. For each chosen basis size, we also compute a ROM based on intrusive projection for comparison with the pOpInf ROM. The intrusive ROM is derived explicitly without any need for approximation of the nonlinear terms via DEIM (see Remark 3.2). Both ROMs are integrated with time step $\delta t = 10^{-3}$ for each parameter in the testing set, and the relative error compared to the full-order model is computed as in (5.1). This procedure yields a single error value for each point in the testing set. The intrusive ROMs are stable throughout the testing set, as are the pOpInf ROMs except at a small number of testing parameters (one for $r_1 = 12, r_2 = 9$; two for $r_1 = r_2 = 4$ and $r_1 = 14$, $r_2 = 11$; and nine for $r_1 = 9$, $r_2 = 7$). At these points, we adjust the time step to $\delta t = 10^{-2}$, producing stable results. For larger basis sizes ($r_1 > 14$, $r_2 > 11$), the pOpInf ROMs are stable throughout the testing set with $\delta t = 10^{-3}$.

Figure 6 shows the 10% quantile, median, 90% quantile, and maximum of the relative errors for each ROM. The median relative error is similar in the training and testing sets and decreases steadily as the basis sizes increase. The pOpInf ROMs generally outperform the intrusive ROMs. The tight interdecile range shows that the ROM error is mostly consistent throughout the parameter space; though not shown, the parameters with higher ROM error are also where the projection error is significantly higher than average. For additional comparison, Figure 7 shows select trajectories of a single pOpInf ROM versus the full-order model. The reduced-order and full-order results are indistinguishable to the eye, which is representative of the ROM performance throughout the testing set.

6. Conclusions. This paper has proposed a non-intrusive parametric model reduction method for parameterized PDEs based on the Operator Inference framework. The approach eliminates the need for interpolation by explicitly learning the reduced

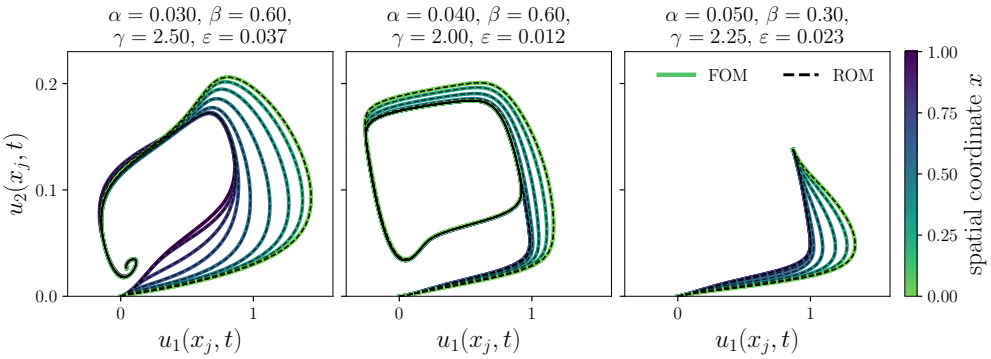


FIG. 7. *Phase portraits of testing trajectories for the FitzHugh-Nagumo problem (3.5a)–(3.5b), traced out at various points in the spatial domain. The solid lines are the full-order trajectories, and the dashed lines are the outputs of the learned pOpInf ROM with* $r_1 = 12$ *and* $r_2 = 9$ *POD modes. The total ROM relative errors in space and time are, from left to right,* 0.034%, 0.777%, *and* 0.009%*; the median relative error on the testing set for* $r_1 = 12$ *and* $r_2 = 9$ *is* 0.014%.

operators of the affine-parametric representation and uses an optimization-based regularization strategy to ensure well-posedness in the learning problem. The parametric ROMs can later be used in outer-loop applications to expedite model evaluations for any choice of the parameters. The efficacy of the method has been demonstrated for two numerical examples: a heat equation with a two-dimensional parametric space, and the FitzHugh-Nagumo system with a four-dimensional parametric space. The resulting ROMs are capable of capturing the behavior of the PDE for parameters outside of the training set and, as shown in the latter example, perform favorably on average when compared to the intrusive ROM with the same affine structure.

It was shown in the FitzHugh-Nagumo example that the learned ROMs successfully capture the inherently different behaviors of the system that come with changes in the parameters (see Figure 7). Yet, as with other data-driven approaches, the quality of inferred ROMs depends strongly on the training set, and one cannot expect a data-driven ROM to produce a particular dynamical behavior that differs wildly from the training data. An important future direction is therefore the automation of an efficient parameter sampling strategy for the offline stage.

Appendix A. Matricization of Tensors. Following the notation of [25], for matrices $\mathbf{W} \in \mathbb{R}^{r\times s}$ and $\mathbf{Z} \in \mathbb{R}^{m\times n}$, let $\mathbf{W} \otimes \mathbf{Z}$ denote the Kronecker product [25, 47]:

$$\mathbf{W} \otimes \mathbf{Z} = \begin{bmatrix} w_{11} & \cdots & w_{1s} \\ \vdots & \ddots & \vdots \\ w_{r1} & \cdots & w_{rs} \end{bmatrix} \otimes \mathbf{Z} := \begin{bmatrix} w_{11}\mathbf{Z} & \cdots & w_{1s}\mathbf{Z} \\ \vdots & \ddots & \vdots \\ w_{r1}\mathbf{Z} & \cdots & w_{rs}\mathbf{Z} \end{bmatrix} \in \mathbb{R}^{rm\times sn},$$

where w_{ij} is the component from the ith row and jth column of $\mathbf{W}$. The definition applies to vectors by setting $s = n = 1$. For $\mathbf{Z} \in \mathbb{R}^{m\times s}$ (i.e., $n = s$), define $\mathbf{W} \odot \mathbf{Z}$ to be the Khatri-Rao product, i.e., the column-wise Kronecker product [23, 25]:

$$\mathbf{W} \odot \mathbf{Z} := \left[\begin{array}{c|c|c|c} \mathbf{w}_1 \otimes \mathbf{z}_1 & \mathbf{w}_2 \otimes \mathbf{z}_2 & \cdots & \mathbf{w}_s \otimes \mathbf{z}_s \end{array}\right] \in \mathbb{R}^{rm\times s},$$

where $\mathbf{w}_i$ and $\mathbf{z}_i$ are the ith columns of $\mathbf{W}$ and $\mathbf{Z}$, respectively.

Let $\mathbf{w} = [w_1 \ \cdots \ w_r] \in \mathbb{R}^r$. As the product $\mathbf{w} \odot \mathbf{w} = \mathbf{w} \otimes \mathbf{w}$ has redundant terms (for instance, $w_1 w_2 = w_2 w_1$ appears twice), we introduce compact second- and

third-order Khatri-Rao products, defined as

$$\mathbf{w} \mathbin{\widehat{\odot}} \mathbf{w} := \begin{bmatrix} w_1^2 \\ w_2(\mathbf{w}_{1:2}) \\ \vdots \\ w_r(\mathbf{w}_{1:r}) \end{bmatrix} \in \mathbb{R}^{\binom{r+1}{2}}, \quad \mathbf{w} \mathbin{\widehat{\odot}} \mathbf{w} \mathbin{\widehat{\odot}} \mathbf{w} := \begin{bmatrix} w_1^3 \\ w_2(\mathbf{w}_{1:2} \mathbin{\widehat{\odot}} \mathbf{w}_{1:2}) \\ \vdots \\ w_r(\mathbf{w}_{1:r} \mathbin{\widehat{\odot}} \mathbf{w}_{1:r}) \end{bmatrix} \in \mathbb{R}^{\binom{r+2}{3}},$$

where $\mathbf{w}_{1:i} = [w_1 \ \cdots \ w_i]^\top \in \mathbb{R}^i$ contains the first i entries of $\mathbf{w}$. Each entry of $\mathbf{w} \mathbin{\widehat{\odot}} \mathbf{w}$ is a product of 2 entries of $\mathbf{w}$, and each product appears exactly once; in other words, $\mathbf{w} \mathbin{\widehat{\odot}} \mathbf{w}$ contains the unique terms of $\mathbf{w}\mathbf{w}^\top$, the 2-fold tensorization of $\mathbf{w}$ [25, 32]. Similarly, each entry of $\mathbf{w} \mathbin{\widehat{\odot}} \mathbf{w} \mathbin{\widehat{\odot}} \mathbf{w}$ is a unique product of 3 entries of $\mathbf{w}$. For matrices, the definition applies column-wise:

$$\left[\begin{array}{c|c|c} \mathbf{w}_1 & \cdots & \mathbf{w}_s \end{array} \right] \mathbin{\widehat{\odot}} \left[\begin{array}{c|c|c} \mathbf{w}_1 & \cdots & \mathbf{w}_s \end{array} \right] = \left[\begin{array}{c|c|c} \mathbf{w}_1 \mathbin{\widehat{\odot}} \mathbf{w}_1 & \cdots & \mathbf{w}_s \mathbin{\widehat{\odot}} \mathbf{w}_s \end{array} \right].$$

With this notation, any linear combination of products of two entries of $\mathbf{w}$ can be represented by the product

$$\mathbf{z}^\top \left(\mathbf{w} \mathbin{\widehat{\odot}} \mathbf{w} \right) = z_1 w_1^2 + z_2 w_1 w_2 + z_3 w_2^2 + z_4 w_1 w_3 + \cdots, \quad \mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \end{bmatrix} \in \mathbb{R}^{\binom{r+1}{2}}.$$

This observation allows us to convert (2.6) to a compact matrix product representation. Specifically, we define $\widehat{\mathbf{H}} \in \mathbb{R}^{r \times \binom{r+1}{2}}$ to be the matrix such that the ith component of the product $\widehat{\mathbf{H}} \left(\widehat{\mathbf{u}} \mathbin{\widehat{\odot}} \widehat{\mathbf{u}} \right)$ is given by

$$[\widehat{\mathbf{H}} \left(\widehat{\mathbf{u}} \mathbin{\widehat{\odot}} \widehat{\mathbf{u}} \right)]_i := \sum_{j=1}^{r} \sum_{l=1}^{r} \langle v_i, \mathcal{H}\left(v_j, v_l; \mu\right) \rangle \, \hat{u}_j \hat{u}_l.$$

Appendix B. Technical Lemma. The following lemma supports Theorem 2.3. Here, $\mathbf{0}_k \in \mathbb{R}^k$ is the vector of k zeros and $\mathbf{1}_k \in \mathbb{R}^k$ denotes the vector of k ones.

LEMMA B.1. *Let $\mathbf{y}_1, \ldots, \mathbf{y}_s \in \mathbb{R}^q$ and $\mathbf{Z}_1, \ldots, \mathbf{Z}_s \in \mathbb{R}^{k \times r}$. Consider the matrix*

$$\mathbf{W} := \begin{bmatrix} \mathbf{y}_1^\top \otimes \mathbf{Z}_1 \\ \vdots \\ \mathbf{y}_s^\top \otimes \mathbf{Z}_s \end{bmatrix} \in \mathbb{R}^{sk \times qr}.$$

If either of the matrices

$$\mathbf{Y} := \begin{bmatrix} \mathbf{y}_1^\top \\ \vdots \\ \mathbf{y}_s^\top \end{bmatrix} \in \mathbb{R}^{s \times q}, \qquad \mathbf{Z} := \begin{bmatrix} \mathbf{Z}_1 \\ \vdots \\ \mathbf{Z}_s \end{bmatrix} \in \mathbb{R}^{sk \times r}$$

do not have full column rank, then neither does $\mathbf{W}$. Conversely, if $\mathbf{Y}$ and <u>each</u> $\mathbf{Z}_1, \ldots, \mathbf{Z}_s$ have full column rank, then so does $\mathbf{W}$.

Proof. Assume that $\mathbf{Y}$ does not have full column rank. Then there exists a nonzero vector $\boldsymbol{\alpha} \in \mathbb{R}^q$ such that $\mathbf{Y}\boldsymbol{\alpha} = \mathbf{0}_s$, that is, $\mathbf{y}_i^\top \boldsymbol{\alpha} = 0$ for $i = 1, \ldots, s$. Using the mixed-product property of the Kronecker product,

$$\left(\mathbf{y}_i^\top \otimes \mathbf{Z}_i\right)(\boldsymbol{\alpha} \otimes \mathbf{1}_r) = \left(\mathbf{y}_i^\top \boldsymbol{\alpha}\right) \otimes (\mathbf{Z}_i \mathbf{1}_r) = 0 \otimes (\mathbf{Z}_i \mathbf{1}_r) = \mathbf{0}_k$$

for $i = 1, \ldots, s$. Then $\mathbf{W}(\boldsymbol{\alpha} \otimes \mathbf{1}_r) = \mathbf{0}_{sk}$. But $\boldsymbol{\alpha} \otimes \mathbf{1}_r$ is a nonzero vector, which implies that the columns of $\mathbf{W}$ are linearly dependent.

Next, suppose that $\mathbf{Z}$ does not have full column rank. Then $\mathbf{Z}\boldsymbol{\beta} = \mathbf{0}_{sk}$ for some nonzero vector $\boldsymbol{\beta} \in \mathbb{R}^r$, implying $\mathbf{Z}_i\boldsymbol{\beta} = \mathbf{0}_k$ for $i = 1, \ldots, s$. Then

$$\left(\mathbf{y}_i^\top \otimes \mathbf{Z}_i\right)(\mathbf{1}_q \otimes \boldsymbol{\beta}) = \left(\mathbf{y}_i^\top \mathbf{1}_q\right) \otimes (\mathbf{Z}_i\boldsymbol{\beta}) = \left(\mathbf{y}_i^\top \mathbf{1}_q\right) \otimes \mathbf{0}_k = \mathbf{0}_k$$

for $i = 1, \ldots, s$, so that $\mathbf{W}(\mathbf{1}_q \otimes \boldsymbol{\beta}) = \mathbf{0}_{sk}$. Since $\mathbf{1}_q \otimes \boldsymbol{\beta}$ is a nonzero vector, $\mathbf{W}$ does not have full column rank.

For the converse statement, assume that $\mathbf{Y}$ and $\mathbf{Z}_1, \ldots, \mathbf{Z}_s$ each have full column rank, and suppose $\boldsymbol{\gamma} = [\boldsymbol{\gamma}_1^\top \ \cdots \ \boldsymbol{\gamma}_q^\top]^\top \in \mathbb{R}^{qr}$, $\boldsymbol{\gamma}_j \in \mathbb{R}^r$, satisfies $\mathbf{W}\boldsymbol{\gamma} = \mathbf{0}_{sk}$, i.e., $(\mathbf{y}_i^\top \otimes \mathbf{Z}_i)\boldsymbol{\gamma} = \mathbf{0}_k$ for $i = 1, \ldots, s$. Denoting $\boldsymbol{\Gamma} = [\boldsymbol{\gamma}_1 \ \cdots \ \boldsymbol{\gamma}_q] \in \mathbb{R}^{r \times q}$, we have

$$\mathbf{0}_k = (\mathbf{y}_i^\top \otimes \mathbf{Z}_i)\boldsymbol{\gamma} = \sum_{j=1}^{q} y_{ij}\mathbf{Z}_i\boldsymbol{\gamma}_j = \mathbf{Z}_i\left(\sum_{j=1}^{q} y_{ij}\boldsymbol{\gamma}_j\right) = \mathbf{Z}_i\boldsymbol{\Gamma}\mathbf{y}_i,$$

where y_{ij} is the jth entry of $\mathbf{y}_i$. Since each $\mathbf{Z}_i$ has full column rank, it must be the case that $\boldsymbol{\Gamma}\mathbf{y}_i = \mathbf{0}_r$ for each $i = 1, \ldots, s$, which in turn implies $\mathbf{Y}\boldsymbol{\Gamma}^\top = \left(\boldsymbol{\Gamma}\mathbf{Y}^\top\right)^\top = \mathbf{0}_{s \times r}$. But $\mathbf{Y}$ having full column rank implies $\boldsymbol{\Gamma} = \mathbf{0}_{r \times q}$, hence $\boldsymbol{\gamma} = \mathbf{0}_{qr}$. Thus, the columns of $\mathbf{W}$ are linearly independent, so $\mathbf{W}$ has full column rank. ∎

Appendix C. General Construction for PDE Systems. We provide here a general construction for the affine Operator Inference problem for systems of PDEs to learn reduced-order models of the form (3.3a)–(3.3b). The problem decouples into d instances of (2.13), that is,

$$\min_{\widehat{\mathbf{O}}_\ell} \left\| \mathbf{D}_\ell \widehat{\mathbf{O}}_\ell^\top - \mathbf{R}_\ell^\top \right\|_F^2, \qquad \ell = 1, \ldots, d,$$

where

$$\mathbf{D}_\ell = \left[\begin{array}{c|c|ccc|cccc} \mathbf{D}_{c_\ell} & \mathbf{D}_{A_{\ell,1}} & \cdots & \mathbf{D}_{A_{\ell,d}} & \mathbf{D}_{H_{\ell,11}} & \mathbf{D}_{H_{\ell,12}} & \cdots & \mathbf{D}_{H_{\ell,dd}} \end{array}\right],$$
$$\widehat{\mathbf{O}}_\ell = \left[\begin{array}{c|ccc|cccc} \widehat{\boldsymbol{\mathcal{C}}}_\ell & \widehat{\boldsymbol{\mathcal{A}}}_{\ell,1} & \cdots & \widehat{\boldsymbol{\mathcal{A}}}_{\ell,d} & \widehat{\boldsymbol{\mathcal{H}}}_{\ell,11} & \widehat{\boldsymbol{\mathcal{H}}}_{\ell,12} & \cdots & \widehat{\boldsymbol{\mathcal{H}}}_{\ell,dd} \end{array}\right],$$
$$\mathbf{R}_\ell = \left[\begin{array}{ccc} \dot{\widehat{\mathbf{U}}}_\ell(\mu_1) & \cdots & \dot{\widehat{\mathbf{U}}}_\ell(\mu_s) \end{array}\right] \in \mathbb{R}^{r_\ell \times sK},$$

with

$$\mathbf{D}_{c_\ell} = \begin{bmatrix} \boldsymbol{\theta}_{c_\ell}(\mu_1) \otimes \mathbf{1}_K \\ \vdots \\ \boldsymbol{\theta}_{c_\ell}(\mu_s) \otimes \mathbf{1}_K \end{bmatrix} \in \mathbb{R}^{K \times q_{c_\ell}},$$
$$\mathbf{D}_{A_{\ell,m}} = \begin{bmatrix} \boldsymbol{\theta}_{A_{\ell,m}}(\mu_1) \otimes \widehat{\mathbf{U}}_m(\mu_1)^\top \\ \vdots \\ \boldsymbol{\theta}_{A_{\ell,m}}(\mu_s) \otimes \widehat{\mathbf{U}}_m(\mu_s)^\top \end{bmatrix} \in \mathbb{R}^{K \times q_{A_{\ell,m}} r_m},$$

$$\mathbf{D}_{H_{\ell,mm}} = \begin{bmatrix} \boldsymbol{\theta}_{H_{\ell,mm}}(\mu_1) \otimes \left(\widehat{\mathbf{U}}_m(\mu_1) \mathbin{\widehat{\odot}} \widehat{\mathbf{U}}_m(\mu_1)\right)^\top \\ \vdots \\ \boldsymbol{\theta}_{H_{\ell,mm}}(\mu_s) \otimes \left(\widehat{\mathbf{U}}_m(\mu_s) \mathbin{\widehat{\odot}} \widehat{\mathbf{U}}_m(\mu_s)\right)^\top \end{bmatrix} \in \mathbb{R}^{K \times q_{H_{\ell,mm}} r_m(r_m+1)/2},$$

$$\mathbf{D}_{H_{\ell,mn}} = \begin{bmatrix} \boldsymbol{\theta}_{H_{\ell,mn}}(\mu_1) \otimes \left(\widehat{\mathbf{U}}_m(\mu_1) \odot \widehat{\mathbf{U}}_n(\mu_1)\right)^\top \\ \vdots \\ \boldsymbol{\theta}_{H_{\ell,mn}}(\mu_s) \otimes \left(\widehat{\mathbf{U}}_m(\mu_s) \odot \widehat{\mathbf{U}}_n(\mu_s)\right)^\top \end{bmatrix} \in \mathbb{R}^{K \times q_{H_{\ell,mn}} r_m r_n}, \quad (n \neq m)$$

$$\widehat{\boldsymbol{\mathcal{C}}}_\ell = \begin{bmatrix} \widehat{\mathbf{c}}_\ell^{(1)} & \cdots & \widehat{\mathbf{c}}_\ell^{(q_{c_\ell})} \end{bmatrix} \in \mathbb{R}^{r_\ell \times q_{c_\ell}}$$

$$\widehat{\boldsymbol{\mathcal{A}}}_{\ell,m} = \begin{bmatrix} \widehat{\mathbf{A}}_{\ell,m}^{(1)} & \cdots & \widehat{\mathbf{A}}_{\ell,m}^{(q_{A_{\ell,m}})} \end{bmatrix} \in \mathbb{R}^{r_\ell \times q_{A_{\ell,m}} r_m}$$

$$\widehat{\boldsymbol{\mathcal{H}}}_{\ell,mm} = \begin{bmatrix} \widehat{\mathbf{H}}_{\ell,mm}^{(1)} & \cdots & \widehat{\mathbf{H}}_{\ell,mm}^{(q_{H_{\ell,mm}})} \end{bmatrix} \in \mathbb{R}^{r_\ell \times q_{H_{\ell,mm}} r_m(r_m+1)/2}$$

$$\widehat{\boldsymbol{\mathcal{H}}}_{\ell,mn} = \begin{bmatrix} \widehat{\mathbf{H}}_{\ell,mn}^{(1)} & \cdots & \widehat{\mathbf{H}}_{\ell,mn}^{(q_{H_{\ell,mn}})} \end{bmatrix} \in \mathbb{R}^{r_\ell \times q_{H_{\ell,mn}} r_m r_n}, \quad (m \neq n)$$

$$\widehat{\mathbf{U}}_\ell(\mu_i) = \begin{bmatrix} \widehat{\mathbf{u}}_\ell(t_1;\mu_i) & \cdots & \widehat{\mathbf{u}}_\ell(t_K;\mu_i) \end{bmatrix} \in \mathbb{R}^{r_\ell \times K},$$

$$\dot{\widehat{\mathbf{U}}}_\ell(\mu_i) = \begin{bmatrix} \frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{u}}_\ell(t;\mu_i)\Big|_{t=t_1} & \cdots & \frac{\mathrm{d}}{\mathrm{d}t}\widehat{\mathbf{u}}_\ell(t;\mu_i)\Big|_{t=t_K} \end{bmatrix} \in \mathbb{R}^{r_\ell \times K},$$

$$\boldsymbol{\theta}_{c_\ell}(\mu_i) = \begin{bmatrix} \theta_{c_\ell}^{(1)}(\mu_i) & \cdots & \theta_{c_\ell}^{(q_{c_\ell})}(\mu_i) \end{bmatrix} \in \mathbb{R}^{1 \times q_{c_\ell}},$$

$$\boldsymbol{\theta}_{A_{\ell,m}}(\mu_i) = \begin{bmatrix} \theta_{A_{\ell,m}}^{(1)}(\mu_i) & \cdots & \theta_{A_{\ell,m}}^{(q_{A_{\ell,m}})}(\mu_i) \end{bmatrix} \in \mathbb{R}^{1 \times q_{A_{\ell,m}}},$$

$$\boldsymbol{\theta}_{H_{\ell,mn}}(\mu_i) = \begin{bmatrix} \theta_{H_{\ell,mn}}^{(1)}(\mu_i) & \cdots & \theta_{H_{\ell,mn}}^{(q_{H_{\ell,mn}})}(\mu_i) \end{bmatrix} \in \mathbb{R}^{1 \times q_{H_{\ell,mn}}}.$$

In practice, this construction is often sparse due to the limited number of terms in the governing PDE.

Acknowledgements. This work has been supported in part by the US Department of Energy AEOLUS MMICC center under award DE-SC0019303, program manager W. Spotz; by the Air Force Center of Excellence on Multi-Fidelity Modeling of Rocket Combustor Dynamics under award FA9550-17-1-0195; and by the US Department of Energy National Nuclear Security Administration under award DE-NA0003969. The authors also wish to thank Elizabeth Qian and Vincent Martinez for their insightful comments.

REFERENCES

[1] V. Algazi and D. Sakrison, *On the optimality of the Karhunen-Loève expansion*, IEEE Transactions on Information Theory, 15 (1969), pp. 319–321.

[2] A. C. Antoulas and B. D. Q. Anderson, *On the scalar rational interpolation problem*, IMA Journal of Mathematical Control and Information, 3 (1986), pp. 61–88.

[3] A. C. Antoulas, I. V. Gosea, and A. C. Ionita, *Model reduction of bilinear systems in the Loewner framework*, SIAM Journal on Scientific Computing, 38 (2016), pp. B889–B916.

[4] H. Arbabi, J. E. Bunder, G. Samaey, A. J. Roberts, and I. G. Kevrekidis, *Linking machine learning with multiscale numerics: Data-driven discovery of homogenized equations*, JOM, 72 (2020), pp. 4444–4457.

[5] U. M. Ascher, S. J. Ruuth, and B. T. R. Wetton, *Implicit-explicit methods for time-dependent partial differential equations*, SIAM Journal on Numerical Analysis, 32 (1995), pp. 797–823.

[6] S. M. Baer and T. Erneux, *Singular Hopf bifurcation to relaxation oscillations*, SIAM Journal on Applied Mathematics, 46 (1986), pp. 721–739.
[7] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera, *An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations*, Comptes Rendus Mathematique, 339 (2004), pp. 667–672.
[8] P. Benner, P. Goyal, B. Kramer, B. Peherstorfer, and K. Willcox, *Operator inference for non-intrusive model reduction of systems with non-polynomial nonlinear terms*, Computer Methods in Applied Mechanics and Engineering, 372 (2020), p. 113433.
[9] P. Benner, S. Gugercin, and K. Willcox, *A survey of projection-based model reduction methods for parametric dynamical systems*, SIAM Review, 57 (2015), pp. 483–531.
[10] G. Berkooz, P. Holmes, and J. Lumley, *The proper orthogonal decomposition in the analysis of turbulent flows*, Annual Review of Fluid Mechanics, 25 (1993), pp. 539–575.
[11] K. Bhattacharya, B. Hosseini, N. B. Kovachki, and A. M. Stuart, *Model reduction and neural networks for parametric PDEs*, The SMAI Journal of Computational Mathematics, 7 (2021), pp. 121–157.
[12] Å. Björck, *Numerical Methods for Least Squares Problems*, SIAM, Philadelphia, PA, 1996.
[13] S. L. Brunton, J. L. Proctor, and J. N. Kutz, *Discovering governing equations from data by sparse identification of nonlinear dynamical systems*, Proceedings of the National Academy of Sciences, 113 (2016), pp. 3932–3937.
[14] S. L. Brunton, J. L. Proctor, J. H. Tu, and J. N. Kutz, *Compressed sensing and dynamic mode decomposition*, Journal of Computational Dynamics, 2 (2015), p. 165.
[15] S. Chaturantabut and D. C. Sorensen, *Nonlinear model reduction via discrete empirical interpolation*, SIAM Journal on Scientific Computing, 32 (2010), pp. 2737–2764.
[16] R. FitzHugh, *Impulses and physiological states in theoretical models of nerve membrane*, Biophysical Journal, 1 (1961), pp. 445–466.
[17] O. Ghattas and K. Willcox, *Learning physics-based models from data: perspectives from inverse problems and model reduction*, Acta Numerica, 30 (2021), pp. 445–554.
[18] G. H. Golub and C. F. Van Loan, *Matrix Computations*, vol. 3, John Hopkins University Press, Baltimore, MD, 2013.
[19] W. R. Graham, J. Peraire, and K. Y. Tang, *Optimal control of vortex shedding using low-order models. Part I—Open-loop model development*, International Journal for Numerical Methods in Engineering, 44 (1999), pp. 945–972.
[20] M. Grepl and A. Patera, *A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations*, ESAIM: Mathematical Modelling and Numerical Analysis, 39 (2005), pp. 157–181.
[21] A. C. Ionita and A. C. Antoulas, *Data-driven parametrized model reduction in the Loewner framework*, SIAM Journal on Scientific Computing, 36 (2014), pp. A984–A1007.
[22] P. Jain, S. McQuarrie, and B. Kramer, *Performance comparison of data-driven reduced models for a single-injector combustion process*, in AIAA Propulsion and Energy 2021 Forum, Virtual Event, 2021. Paper AIAA-2021-3633.
[23] C. Khatri and C. R. Rao, *Solutions to some functional equations and their applications to characterization of probability distributions*, Sankhyā: The Indian Journal of Statistics, Series A, (1968), pp. 167–180.
[24] P. Khodabakhshi and K. E. Willcox, *Non-intrusive data-driven model reduction for differential algebraic equations derived from lifting transformations*, Oden Institute Report 21-08, University of Texas at Austin, 2021.
[25] T. G. Kolda and B. W. Bader, *Tensor decompositions and applications*, SIAM Review, 51 (2009), pp. 455–500.
[26] K. Lee and K. T. Carlberg, *Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders*, Journal of Computational Physics, 404 (2020), p. 108973.
[27] S. Lee, M. Kooshkbaghi, K. Spiliotis, C. I. Siettos, and I. G. Kevrekidis, *Coarse-scale PDEs from fine-scale observations via machine learning*, Chaos: An Interdisciplinary Journal of Nonlinear Science, 30 (2020), p. 013141.
[28] J. Lumley, *The structures of inhomogeneous turbulent flow*, Atmospheric Turbulence and Radio Wave Propagation, (1967), pp. 166–178.
[29] S. A. McQuarrie, C. Huang, and K. E. Willcox, *Data-driven reduced-order models via regularised operator inference for a single-injector combustion process*, Journal of the Royal Society of New Zealand, 51 (2021), pp. 194–211.
[30] J. Nagumo, S. Arimoto, and S. Yoshizawa, *An active pulse transmission line simulating nerve axon*, Proceedings of the IRE, 50 (1962), pp. 2061–2070.
[31] J. A. Nelder and R. Mead, *A simplex method for function minimization*, The Computer

Journal, 7 (1965), pp. 308–313.
[32] G. Ongie, D. Pimentel-Alarcón, L. Balzano, R. Willett, and R. D. Nowak, *Tensor methods for nonlinear matrix completion*, SIAM Journal on Mathematics of Data Science, 3 (2021), pp. 253–279.
[33] B. Peherstorfer, *Sampling low-dimensional Markovian dynamics for pre-asymptotically recovering reduced models from data with operator inference*, SIAM Journal on Scientific Computing, 42 (2020), pp. A3489–A3515.
[34] B. Peherstorfer and K. Willcox, *Data-driven operator inference for nonintrusive projection-based model reduction*, Computer Methods in Applied Mechanics and Engineering, 306 (2016), pp. 196–215.
[35] E. Qian, B. Kramer, B. Peherstorfer, and K. Willcox, *Lift & Learn: Physics-informed machine learning for large-scale nonlinear dynamical systems.*, Physica D: Nonlinear Phenomena, 406 (2020), p. 132401.
[36] M. Quade, M. Abel, J. Nathan Kutz, and S. L. Brunton, *Sparse identification of nonlinear dynamics for rapid model recovery*, Chaos: An Interdisciplinary Journal of Nonlinear Science, 28 (2018), p. 063116.
[37] G. Rozza, D. Huynh, and A. Patera, *Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics*, Archives of Computational Methods in Engineering, 15 (2008), pp. 229–275.
[38] S. Rudy, A. Alla, S. L. Brunton, and J. N. Kutz, *Data-driven identification of parametric partial differential equations*, SIAM Journal on Applied Dynamical Systems, 18 (2019), pp. 643–660.
[39] H. Schaeffer, *Learning partial differential equations via data discovery and sparse optimization*, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 473 (2017), p. 20160446.
[40] P. J. Schmid, *Dynamic mode decomposition of numerical and experimental data*, Journal of Fluid Mechanics, 656 (2010), pp. 5–28.
[41] L. Sirovich, *Turbulence and the dynamics of coherent structures. I. Coherent structures*, Quarterly of Applied Mathematics, 45 (1987), pp. 561–571.
[42] R. Swischuk, B. Kramer, C. Huang, and K. Willcox, *Learning physics-based reduced-order models for a single-injector combustion process*, AIAA Journal, 58 (2020), pp. 2658–2672.
[43] M. Tezzele, N. Demo, G. Stabile, A. Mola, and G. Rozza, *Enhancing CFD predictions in shape design problems by model and parameter space reduction*, Advanced Modeling and Simulation in Engineering Sciences, 7 (2020), pp. 1–19.
[44] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Inverse Problems*, Wiley, New York, NY, 1977.
[45] L. N. Trefethen and D. Bau III, *Numerical Linear Algebra*, SIAM, Philadelphia, PA, 1997.
[46] W. I. T. Uy and B. Peherstorfer, *Probabilistic error estimation for non-intrusive reduced models learned from data of systems governed by linear parabolic partial differential equations*, ESAIM: Mathematical Modelling and Numerical Analysis, 55 (2021), pp. 735–761.
[47] C. F. Van Loan, *The ubiquitous Kronecker product*, Journal of Computational and Applied Mathematics, 123 (2000), pp. 85–100.
[48] K. Veroy and A. Patera, *Certified real-time solution of the parametrized steady incompressible Navier-Stokes equations: Rigorous reduced-basis a posteriori error bounds*, International Journal for Numerical Methods in Fluids, 47 (2005), pp. 773–788.
[49] K. Veroy, C. Prud'homme, D. Rovas, and A. Patera, *A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations*, in Proceedings of the 16th AIAA Computational Fluid Dynamics Conference, Orlando, FL, 2003. Paper AIAA-2003-3847.
[50] S. Yildiz, P. Goyal, P. Benner, and B. Karasözen, *Learning reduced-order dynamics for parametrized shallow water equations from data*, International Journal for Numerical Methods in Fluids, 93 (2021), pp. 2803–2821.